Red Hat Enterprise Linux 6
Virtual Server Administration

Lastverteilungs-Add-On für Red Hat Enterprise Linux
Ausgabe 6

Landmann

Red Hat Enterprise Linux 6 Virtual Server Administration

Lastverteilungs-Add-On für Red Hat Enterprise Linux
Ausgabe 6

Landmann
rlandmann@redhat.com

Rechtlicher Hinweis

Copyright © 2010 Red Hat, Inc.

This document is licensed by Red Hat under the Creative Commons Attribution-ShareAlike 3.0 Unported License. If you distribute this document, or a modified version of it, you must provide attribution to Red Hat, Inc. and provide a link to the original. If the document is modified, all Red Hat trademarks must be removed.

Red Hat, as the licensor of this document, waives the right to enforce, and agrees not to assert, Section 4d of CC-BY-SA to the fullest extent permitted by applicable law.

Red Hat, Red Hat Enterprise Linux, the Shadowman logo, JBoss, MetaMatrix, Fedora, the Infinity Logo, and RHCE are trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux ® is the registered trademark of Linus Torvalds in the United States and other countries.

Java ® is a registered trademark of Oracle and/or its affiliates.

XFS ® is a trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or other countries.

MySQL ® is a registered trademark of MySQL AB in the United States, the European Union and other countries.

Node.js ® is an official trademark of Joyent. Red Hat Software Collections is not formally related to or endorsed by the official Joyent Node.js open source or commercial project.

The OpenStack ® Word Mark and OpenStack Logo are either registered trademarks/service marks or trademarks/service marks of the OpenStack Foundation, in the United States and other countries and are used with the OpenStack Foundation's permission. We are not affiliated with, endorsed or sponsored by the OpenStack Foundation, or the OpenStack community.

All other trademarks are the property of their respective owners.

Zusammenfassung

Das Erstellen eines Lastverteilungs-Add-On Systems bietet eine hochverfügbare und skalierbare Lösung für Produktionsdienste, die spezialisierte Linux Virtual Servers (LVS) für das Routing und für die Lastverteilung verwenden und die mit Hilfe von PIRANHA konfiguriert werden. Dieses Buch behandelt die Konfiguration von Hochverfügbarkeitssystemen und -diensten mit Red Hat Enterprise Linux und dem Lastverteilungs-Add-On für Red Hat Enterprise Linux 6.

Inhaltsverzeichnis

Einführung **4**
1. Dokumentkonventionen 4
1.1. Typografische Konventionen 5
1.2. Konventionen für Seitenansprachen 6
1.3. Anmerkungen und Warnungen 7
2. Feedback 8

Kapitel 1. Überblick über das Lastverteilungs-Add-On **9**
1.1. Eine grundlegende Lastverteilungs-Add-On-Konfiguration 9
1.1.1. Datenreplikation und Datenaustausch zwischen realen Servern 11
1.1.1.1. Konfiguration von realen Servern zur Synchronisation von Daten 11
1.2. Eine Three-Tier-Lastverteilungs-Add-On-Konfiguration 11
1.3. Lastverteilungs-Add-On Scheduling-Überblick 12
1.3.1. Scheduling-Algorithmen 13
1.3.2. Server-Gewichtung und Scheduling 14
1.4. Routing-Methoden 15
1.4.1. NAT-Routing 15
1.4.2. Direktes Routing 16
1.4.2.1. Direktes Routing und die ARP-Einschränkung 17
1.5. Persistenz und Firewall-Markierungen 18
1.5.1. Persistenz 18
1.5.2. Firewall-Markierungen 18
1.6. Lastverteilungs-Add-On — ein Block-Diagramm 19
1.6.1. Komponenten des Lastverteilungs-Add-Ons 20
1.6.1.1. pulse 20
1.6.1.2. lvs 20
1.6.1.3. ipvsadm 20
1.6.1.4. nanny 20
1.6.1.5. /etc/sysconfig/ha/lvs.cf 20
1.6.1.6. Piranha Configuration Tool 21
1.6.1.7. send_arp 21

Kapitel 2. Erstmalige Konfiguration des Lastverteilungs-Add-Ons **22**
2.1. Konfiguration von Diensten auf den LVS-Routern 22
2.2. Einrichten eines Passworts für das Piranha Configuration Tool 23
2.3. Starten des Piranha Configuration Tool Dienstes 23
2.3.1. Konfiguration des Web-Server-Ports für das Piranha Configuration Tool 24
2.4. Zugangsbeschränkung zum Piranha Configuration Tool 24
2.5. Aktivierung der Paketweiterleitung 25
2.6. Konfiguration von Diensten auf den realen Servern 26

Kapitel 3. Einrichten des Lastverteilungs-Add-Ons **27**
3.1. Das NAT Lastverteilungs-Add-On-Netzwerk 27
3.1.1. Konfiguration von Netzwerkschnittstellen für das Lastverteilungs-Add-On mit NAT 27
3.1.2. Routing auf den realen Servern 28
3.1.3. Aktivierung von NAT-Routing auf den LVS-Routern 29
3.2. Das Lastverteilungs-Add-On via direktem Routing 30
3.2.1. Direktes Routing und arptables_jf 31
3.2.2. Direktes Routing und iptables 32
3.3. Zusammensetzen der Konfiguration 33
3.3.1. Allgemeine Tipps bezüglich Lastverteilungs-Add-On-Vernetzung 33
3.4. Multiport-Dienste und das Lastverteilungs-Add-On 34

3.4.1. Zuweisen von Firewall-Markierungen 34
3.5. Konfiguration von FTP 35
3.5.1. Funktionsweise von FTP 35
3.5.2. Auswirkungen auf das Lastverteilungs-Add-On-Routing 36
3.5.3. Erstellen von Netzwerk-Paketfilterregeln 36
3.5.3.1. Regeln für aktive Verbindungen 36
3.5.3.2. Regeln für passive Verbindungen 37
3.6. Speichern der Einstellungen des Netzwerkpaketfilters 38

Kapitel 4. Konfiguration des Lastverteilungs-Add-Ons mit dem Piranha Configuration Tool 39
4.1. Benötigte Software 39
4.2. Einloggen in das Piranha Configuration Tool 39
4.3. CONTROL/MONITORING 40
4.4. GLOBAL SETTINGS 41
4.5. REDUNDANCY 43
4.6. VIRTUAL SERVERS 45
4.6.1. Der Unterabschnitt VIRTUAL SERVER 46
4.6.2. Der Unterabschnitt REAL SERVER 49
4.6.3. Der Unterabschnitt EDIT MONITORING SCRIPTS 51
4.7. Synchronisation von Konfigurationsdateien 53
4.7.1. Synchronisation von lvs.cf 53
4.7.2. Synchronisation von sysctl 54
4.7.3. Synchronisation von Regeln zur Netzwerk-Paket-Filterung 54
4.8. Das Lastverteilungs-Add-On starten 54

Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On 56

Revisionsverlauf 58

Stichwortverzeichnis 58
Symbole 58
A 58
C 58
D 58
E 58
F 58
G 58
H 59
I 59
J 59
K 59
L 59
M 60
N 60
P 61
R 61
S 62

Einführung

Dieses Dokument liefert Informationen zur Installation, Konfiguration und Verwaltung der Lastverteilungs-Add-On Komponenten. Das Lastverteilungs-Add-On bietet Lastverteilung durch spezialisierte Routing-Techniken, die Datenverkehr an einen Server-Pool senden.

Die Zielgruppe dieses Dokuments sollte fortgeschrittene praktische Erfahrungen im Umgang mit Red Hat Enterprise Linux besitzen und die Begriffe Cluster, Storage (Speicher) und Server-Computing verstehen.

Dieses Dokument ist wie folgt aufgebaut:

- Kapitel 1, *Überblick über das Lastverteilungs-Add-On*
- Kapitel 2, *Erstmalige Konfiguration des Lastverteilungs-Add-Ons*
- Kapitel 3, *Einrichten des Lastverteilungs-Add-Ons*
- Kapitel 4, *Konfiguration des Lastverteilungs-Add-Ons mit dem **Piranha Configuration Tool***
- Anhang A, *Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On*

Für weitere Informationen zu Red Hat Enterprise Linux 6, werfen Sie einen Blick auf die folgenden Quellen:

- *Red Hat Enterprise Linux Installationshandbuch* — Liefert Informationen bezüglich der Installation von Red Hat Enterprise Linux 6.
- *Red Hat Enterprise Linux Bereitstellungshandbuch* — Liefert Informationen bezüglich des Einsatzes, der Konfiguration und der Administration von Red Hat Enterprise Linux 6.

Werfen Sie einen Blick auf die folgenden Quellen für weitere Informationen zum Lastverteilungs-Add-On für Red Hat Enterprise Linux 6:

- *Red Hat Cluster Suite Überblick* — Liefert einen Überblick über das Hochverfügbarkeits-Add-On, das Resilient-Storage-Add-On und das Lastverteilungs-Add-On auf hohem Niveau.
- *Konfiguration und Verwaltung des Hochverfügbarkeits-Add-Ons* — Liefert Informationen bezüglich der Konfiguration und der Verwaltung des Hochverfügbarkeits-Add-Ons (auch als Red Hat Cluster bekannt) für Red Hat Enterprise Linux 6.
- *Logical Volume Manager Administration* — Liefert eine Beschreibung des Logical Volume Manager (LVM), inklusive der Informationen für den Betrieb von LVM in einem Cluster-Umgebung.
- *Global File System 2: Konfiguration und Administration* — Liefert Informationen bezüglich der Installation, der Konfiguration und der Wartung von Red Hat Resilient-Storage-Add-Ons (auch als Red Hat Global File System 2 bekannt).
- *DM-Multipath* — Liefert Informationen bezüglich des Device-Mapper Multipath Features von Red Hat Enterprise Linux 6.
- *Versionshinweise* — Liefert Informationen zu aktuellen Releases von Red Hat Produkten.

Dieses und weitere Red Hat-Dokumente stehen als HTML-, PDF- und RPM-Versionen auf der Red Hat Enterprise Linux-Dokumentations-CD und online unter http://www.redhat.com/docs/ zur Verfügung.

1. Dokumentkonventionen

Dieses Handbuch verwendet mehrere Konventionen, um bestimmte Wörter und Sätze hervorzuheben und Aufmerksamkeit auf bestimmte Informationen zu lenken.

In PDF- und Papierausgaben verwendet dieses Handbuch Schriftbilder des Liberation-Fonts-Sets. Das Liberation-Fonts-Set wird auch für HTML-Ausgaben verwendet, falls es auf Ihrem System installiert ist.

Falls nicht, werden alternative, aber äquivalente Schriftbilder angezeigt. Beachten Sie: Red Hat Enterprise Linux 5 und die nachfolgende Versionen beinhalten das Liberation-Fonts-Set standardmäßig.

1.1. Typografische Konventionen

Es werden vier typografische Konventionen verwendet, um die Aufmerksamkeit auf bestimmte Wörter und Sätze zu lenken. Diese Konventionen und die Umstände, unter denen sie auftreten, sind folgende:

`Nichtproportional Fett`

Dies wird verwendet, um Systemeingaben hervorzuheben, einschließlich Shell-Befehle, Dateinamen und -pfade. Es wird ebenfalls zum Hervorheben von Tasten und Tastenkombinationen verwendet. Zum Beispiel:

> Um den Inhalt der Datei **`my_next_bestselling_novel`** in Ihrem aktuellen Arbeitsverzeichnis zu sehen, geben Sie den Befehl **`cat my_next_bestselling_novel`** in den Shell-Prompt ein und drücken Sie **`Enter`**, um den Befehl auszuführen.

Das oben aufgeführte Beispiel beinhaltet einen Dateinamen, einen Shell-Befehl und eine Taste. Alle werden nichtproportional fett dargestellt und alle können, dank des Kontextes, leicht unterschieden werden.

Tastenkombinationen unterscheiden sich von einzelnen Tasten durch das Pluszeichen, das die einzelnen Teile einer Tastenkombination miteinander verbindet. Zum Beispiel:

> Drücken Sie **`Enter`**, um den Befehl auszuführen.
>
> Drücken Sie **`Strg`**+**`Alt`**+**`F2`**, um zu einem virtuellen Terminal zu wechseln.

Das erste Beispiel hebt die zu drückende Taste hervor. Das zweite Beispiel hebt eine Tastenkombination hervor: eine Gruppe von drei Tasten, die gleichzeitig gedrückt werden müssen.

Falls Quellcode diskutiert wird, werden Klassennamen, Methoden, Funktionen, Variablennamen und Rückgabewerte, die innerhalb eines Abschnitts erwähnt werden, wie oben gezeigt **`nichtproportional fett`** dargestellt. Zum Beispiel:

> Zu dateiverwandten Klassen zählen **`filesystem`** für Dateisysteme, **`file`** für Dateien und **`dir`** für Verzeichnisse. Jede Klasse hat ihren eigenen Satz an Berechtigungen.

Proportional Fett

Dies kennzeichnet Wörter oder Sätze, die auf einem System vorkommen, einschließlich Applikationsnamen, Text in Dialogfeldern, beschriftete Schaltflächen, Bezeichnungen für Auswahlkästchen und Radio-Buttons, Überschriften von Menüs und Untermenüs. Zum Beispiel:

> Wählen Sie **System** → **Einstellungen** → **Maus** in der Hauptmenüleiste aus, um die **Mauseinstellungen** zu öffnen. Wählen Sie im Reiter **`Tasten`** auf das Auswahlkästchen **`Mit links bediente Maus`** und anschließend auf **`Schließen`**, um die primäre Maustaste von der linken auf die rechte Seite zu ändern (d.h., um die Maus auf Linkshänder anzupassen).
>
> Um ein Sonderzeichen in eine **`gedit`**-Datei einzufügen, wählen Sie **Anwendungen** → **Zubehör** → **Zeichentabelle** aus der Hauptmenüleiste. Wählen Sie als Nächstes **Suchen** → **Suchen** aus der Menüleiste der **`Zeichentabelle`**, geben Sie im Feld **`Suchbegriff`** den Namen des Zeichens ein und klicken Sie auf **`Weitersuchen`**. Das gesuchte Zeichen wird daraufhin in der **`Zeichentabelle`** hervorgehoben. Doppelklicken Sie auf dieses

hervorgehobene Zeichen, um es in das Feld **Zu kopierender Text** zu übernehmen und klicken Sie anschließend auf die Schaltfläche **Kopieren**. Gehen Sie nun zurück in Ihr Dokument und wählen Sie **Bearbeiten** → **Einfügen** aus der **gedit**-Menüleiste.

Der oben aufgeführte Text enthält Applikationsnamen, systemweite Menünamen und -elemente, applikationsspezifische Menünamen sowie Schaltflächen und Text innerhalb einer grafischen Oberfläche. Alle werden proportional fett dargestellt und sind anhand des Kontextes unterscheidbar.

Nichtproportional Fett Kursiv oder ***Proportional Fett Kursiv***

Sowohl bei nichtproportional fett als auch bei proportional fett weist ein zusätzlicher Kursivdruck auf einen ersetzbaren oder variablen Text hin. Kursivdruck kennzeichnet Text, der nicht wörtlich eingeben wird, oder angezeigten Text, der sich abhängig von den gegebenen Umständen unterscheiden kann. Zum Beispiel:

Um sich mit einer Remote-Maschine via SSH zu verbinden, geben Sie an einem Shell-Prompt **ssh *username*@*domain.name*** ein. Falls die Remote-Maschine **example.com** ist und Ihr Benutzername auf dieser Maschine John lautet, geben Sie also **ssh john@example.com** ein.

Der Befehl **mount -o remount *file-system*** hängt das angegebene Dateisystem wieder ein. Um beispielsweise das **/home**-Dateisystem wieder einzuhängen, verwenden Sie den Befehl **mount -o remount /home**.

Um die Version des derzeit installierten Pakets zu sehen, verwenden Sie den Befehl **rpm -q *package***. Die Ausgabe sieht wie folgt aus: ***package-version-release***.

Beachten Sie die kursiv dargestellten Begriffe oben — username, domain.name, file-system, package, version und release. Jedes Wort ist ein Platzhalter entweder für Text, den Sie für einen Befehl eingeben, oder für Text, der vom System angezeigt wird.

Neben der Standardbenutzung für die Darstellung des Titels eines Werks zeigt der Kursivdruck auch die erstmalige Verwendung eines neuen und wichtigen Begriffs an. Zum Beispiel:

Publican ist ein *DocBook* Publishing-System.

1.2. Konventionen für Seitenansprachen

Ausgaben des Terminals und Auszüge aus dem Quellcode werden visuell vom umliegenden Text hervorgehoben durch sogenannte Seitenansprachen (auch Pull-Quotes genannt).

Eine an das Terminal gesendete Ausgabe wird in den Schrifttyp **nichtproportional Roman** gesetzt und wie folgt dargestellt:

```
books        Desktop   documentation  drafts  mss    photos   stuff  svn
books_tests  Desktop1  downloads      images  notes  scripts  svgs
```

Auszüge aus dem Quellcode werden ebenfalls in den Schrifttyp **nichtproportional Roman** gesetzt, doch wird zusätztlich noch die Syntax hervorgehoben:

```
static int kvm_vm_ioctl_deassign_device(struct kvm *kvm,
                  struct kvm_assigned_pci_dev *assigned_dev)
{
          int r = 0;
          struct kvm_assigned_dev_kernel *match;

          mutex_lock(&kvm->lock);

          match = kvm_find_assigned_dev(&kvm->arch.assigned_dev_head,
                                        assigned_dev->assigned_dev_id);
          if (!match) {
                  printk(KERN_INFO "%s: device hasn't been assigned before, "
                    "so cannot be deassigned\n", __func__);
                  r = -EINVAL;
                  goto out;
          }

          kvm_deassign_device(kvm, match);

          kvm_free_assigned_device(kvm, match);

out:
          mutex_unlock(&kvm->lock);
          return r;
}
```

1.3. Anmerkungen und Warnungen

Zu guter Letzt verwenden wir drei visuelle Stile, um die Aufmerksamkeit auf Informationen zu lenken, die andernfalls vielleicht übersehen werden könnten.

Anmerkung

Eine Anmerkung ist ein Tipp, ein abgekürztes Verfahren oder ein alternativer Ansatz für die vorliegende Aufgabe. Das Ignorieren von Anmerkungen sollte keine negativen Auswirkungen haben, aber Sie verpassen so vielleicht einen Trick, der Ihnen das Leben vereinfachen könnte.

Wichtig

Die Wichtig-Schaukästen lenken die Aufmerksamkeit auf Dinge, die sonst leicht übersehen werden können: Konfigurationsänderungen, die nur für die aktuelle Sitzung gelten oder Dienste, für die ein Neustart nötig ist, bevor eine Aktualisierung wirksam wird. Das Ignorieren von Wichtig-Schaukästen würde keinen Datenverlust verursachen, kann aber unter Umständen zu Ärgernissen und Frustration führen.

Warnung

Eine Warnung sollte nicht ignoriert werden. Das Ignorieren von Warnungen führt mit hoher Wahrscheinlichkeit zu Datenverlust.

2. Feedback

Falls Sie einen Druckfehler in diesem Handbuch finden, oder falls Sie sich Gedanken gemacht haben, wie dieses Handbuch verbessert werden könnte, würden wir gerne von Ihnen hören! Bitte reichen Sie einen Fehlerbericht in Bugzilla (http://bugzilla.redhat.com/bugzilla/) für die Komponente **Documentation-cluster** ein.

Vergewissern Sie sich beim Einreichen eines Fehlerberichts, dass Sie die Kennung des Handbuchs mit angeben:

```
Virtual_Server_Administration(EN)-6 (2010-10-14T16:28)
```

Indem Sie die Kennung des Handbuchs angeben, können wir exakt nachvollziehen, um welche Version es sich bei dem Handbuch handelt.

Falls Sie uns einen Vorschlag zur Verbesserung der Dokumentation senden möchten, sollten Sie hierzu möglichst genaue Angaben machen Wenn Sie einen Fehler gefunden haben, geben Sie bitte die Nummer des Abschnitts und einen Ausschnitt des Textes an, damit wir diesen leicht finden können.

Kapitel 1. Überblick über das Lastverteilungs-Add-On

Das Lastverteilungs-Add-On ist ein Set integrierter Software-Komponenten, die Linux Virtual Server (LVS) zur gleichmäßigen Verteilung von IP-Load innerhalb einer Gruppe realer Server liefern. Das Lastverteilungs-Add-On läuft auf einem Paar gleich konfigurierter Computer: Ein Computer, der einen *aktiven LVS-Router* darstellt und ein Computer, der als ein *Backup-LVS-Router* agiert. Der aktive LVS-Router erfüllt zwei Rollen:

- Gleichmäßige Verteilung der Auslastung unter den realen Servern.
- Überprüfung der Integrität der Dienste auf jedem realen Server.

Der Backup-LVS-Router überwacht den aktiven LVS-Router und übernimmt dessen Aufgaben im Falle eines Ausfalls.

Dieses Kapitel liefert einen Überblick über die Lastverteilungs-Add-On-Komponenten und -funktionen und besteht aus den folgenden Abschnitten:

- Abschnitt 1.1, „Eine grundlegende Lastverteilungs-Add-On-Konfiguration“
- Abschnitt 1.2, „Eine Three-Tier-Lastverteilungs-Add-On-Konfiguration“
- Abschnitt 1.3, „Lastverteilungs-Add-On Scheduling-Überblick“
- Abschnitt 1.4, „Routing-Methoden“
- Abschnitt 1.5, „Persistenz und Firewall-Markierungen“
- Abschnitt 1.6, „Lastverteilungs-Add-On — ein Block-Diagramm“

1.1. Eine grundlegende Lastverteilungs-Add-On-Konfiguration

Abbildung 1.1, „Eine grundlegende Lastverteilungs-Add-On-Konfiguration“ zeigt eine einfache Lastverteilungs-Add-On-Konfiguration bestehend aus zwei Schichten. In der ersten Schicht befinden sich zwei LVS-Router — ein aktiver und ein Backup. Jeder der LVS-Router besitzt zwei Netzwerkschnittstellen, eine Schnittstelle im Internet und eine im privaten Netzwerk, die die Regulierung des Datenverkehrs zwischen beiden Netzwerken ermöglichen. In diesem Beispiel verwendet der aktive Router *Network Address Translation* oder *NAT* zur Weiterleitung von Datenverkehr aus dem Internet an eine variable Anzahl an realen Servern in der zweiten Schicht, die im Gegenzug wiederum die notwendigen Dienste zur Verfügung stellen. Daher sind die realen Server in diesem Beispiel mit einem dedizierten privaten Netzwerksegment verbunden und leiten sämtlichen öffentlichen Datenverkehr durch den aktiven LVS-Router rein und raus. Von extern erscheinen die Server als eine Einheit.

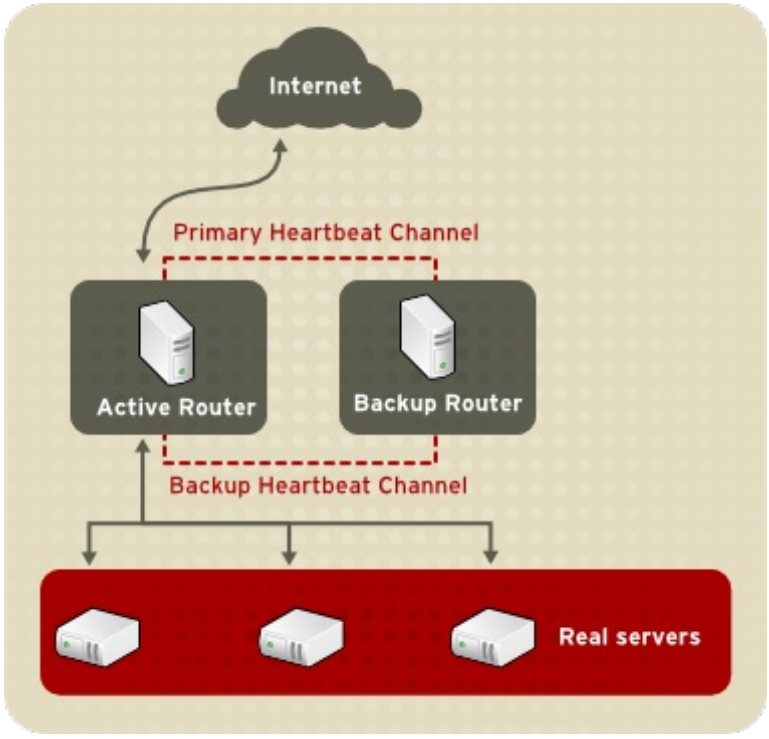

Abbildung 1.1. Eine grundlegende Lastverteilungs-Add-On-Konfiguration

Anfragen an Dienste, die bei einem LVS-Router ankommen, werden an eine *virtuelle IP*-Adresse oder *VIP* adressiert. Dies ist eine öffentlich routbare Adresse, die der Administrator einer Site mit einem vollständigen Namen einer Domain (FQDN) verknüpft, wie z.B. www.example.com und die einem oder mehreren *virtuellen Servern* zugewiesen wird. [1]. Beachten Sie bitte, dass eine VIP-Adresse während einer Ausfallsicherung von einem LVS-Router auf den anderen migriert wird und auf diese Weise eine Präsenz auch weiterhin unter dieser IP-Adresse gewährleistet ist. Dies ist auch als *Floating IP-Adressen* bekannt.

VIP-Adressen können eine Alias-Bezeichnung mit demselben Gerät erhalten, welches den LVS-Router mit dem öffentlichen Netzwerk verbindet. Wenn zum Beispiel eth0 mit dem Internet verbunden ist, dann können mehrere virtuelle Server eine Alias-Bezeichnung **`eth0:1`** erhalten. Alternativ kann jeder virtuelle Server mit einem separaten Gerät pro Dienst verknüpft werden. Beispielsweise kann HTTP-Datenverkehr auf **`eth0:1`** und FTP-Datenverkehr auf **`eth0:2`** verwaltet werden.

Nur jeweils ein LVS-Router ist aktiv. Die Aufgabe des aktiven LVS-Routers ist es, Dienstanfragen von virtuellen IP-Adressen an die realen Server umzuleiten. Die Umleitung basiert auf einem von acht Algorithmen für die Lastverteilung, die näher im Abschnitt 1.3, „Lastverteilungs-Add-On Scheduling-Überblick“ beschrieben werden.

Auch überwacht der aktive Router dynamisch die Gesamtverfassung der speziellen Dienste auf den realen Servern durch einfache *send/expect-Skripte*. Als Hilfe bei der Analyse des Zustands eines Dienstes, der dynamische Daten wie HTTPS oder SSL benötigt, können Sie auch externe ausführbare Dateien aufrufen. Falls ein Dienst auf einem realen Server nicht ordnungsgemäß funktioniert, hört der aktive Router auf, Jobs an diesen Server zu senden, bis dieser wieder auf normalem Betriebslevel ist.

Der Backup-Router übernimmt die Rolle eines Systems in Bereitschaft. Die LVS-Router tauschen periodisch Heartbeat-Meldungen durch die primäre externe öffentliche Schnittstelle, und im Falle einer Ausfallsicherung, durch die private Schnittstelle aus. Erhält der Backup-Knoten keine Heartbeat-Meldung innerhalb eines erwarteten Intervalls, leitet er eine Ausfallsicherung ein und übernimmt die Rolle des aktiven Routers. Während der Ausfallsicherung übernimmt der Backup-Router die VIP-Adressen, die

vom ausgefallenen Router bereitgestellt wurden, unter Verwendung einer Technik, die als *ARP-Spoofing* bekannt ist — hier zeigt der Backup-LVS-Router an, dass er das Ziel für IP-Pakete, die an den ausgefallenen Knoten gerichtet sind, darstellt. Falls der ausgefallene Knoten wieder aktiv wird, nimmt der Backup-Knoten seine Backup-Rolle wieder auf.

Die einfache, zweischichtige Konfiguration unter Abbildung 1.1, „Eine grundlegende Lastverteilungs-Add-On-Konfiguration“ ist am besten zur Bereitstellung von Daten geeignet, die sich nicht sehr häufig ändern — wie beispielsweise statische Web-Seiten — weil die einzelnen realen Server ihre Daten nicht automatisch untereinander synchronisieren.

1.1.1. Datenreplikation und Datenaustausch zwischen realen Servern

Da es keine integrierte Komponente mit dem Lastverteilungs-Add-On gibt, um dieselben Daten unter realen Servern gemeinsam zu nutzen, stehen dem Administrator zwei Grundoptionen zur Verfügung:

- Synchronisation der Daten innerhalb des Pools der realen Server.
- Hinzufügen einer dritten Ebene zur Topologie für den Zugriff auf gemeinsam genutzte Daten.

Die erste Option wird vorzugsweise für Server verwendet, bei denen einer großen Anzahl von Benutzern das Hochladen oder Verändern von Daten auf realen Servern verweigert wird. Falls die Konfiguration einer großen Anzahl von Benutzern gestattet, Daten zu verändern, wie beispielsweise einer E-Commerce-Website, ist das Hinzufügen einer dritten Schicht besser.

1.1.1.1. Konfiguration von realen Servern zur Synchronisation von Daten

Administratoren steht eine Reihe von Möglichkeiten zur Auswahl, um Daten innerhalb des Pools von realen Servern zu synchronisieren. So können beispielsweise Shell-Skripte eingesetzt werden, so dass bei einer Aktualisierung einer Seite durch einen Web-Engineer, diese gleichzeitig an alle Server versendet wird. Auch kann der Systemadministrator Programme wie **rsync** verwenden, um Daten, die sich geändert haben, auf allen Knoten zu einem festgelegten Zeitintervall zu replizieren.

Diese Art der Datensynchronisation funktioniert allerdings nicht optimal, wenn die Konfiguration mit Benutzern überlastet ist, die anhaltend Dateien hochladen oder Datenbanktransaktionen ausführen. Für eine Konfiguration mit hoher Auslastung ist eine *Three-Tiered-Topologie* die ideale Lösung.

1.2. Eine Three-Tier-Lastverteilungs-Add-On-Konfiguration

Abbildung 1.2, „Eine Three-Tier-Lastverteilungs-Add-On-Konfiguration“ zeigt eine typische Three-Tier-Lastverteilungs-Add-On-Topologie. In diesem Beispiel routet der aktive LVS-Router die Anfragen aus dem Internet an den Pool der realen Server weiter. Jeder der realen Server greift dann via Netzwerk auf eine gemeinsam genutzte Datenquelle zu.

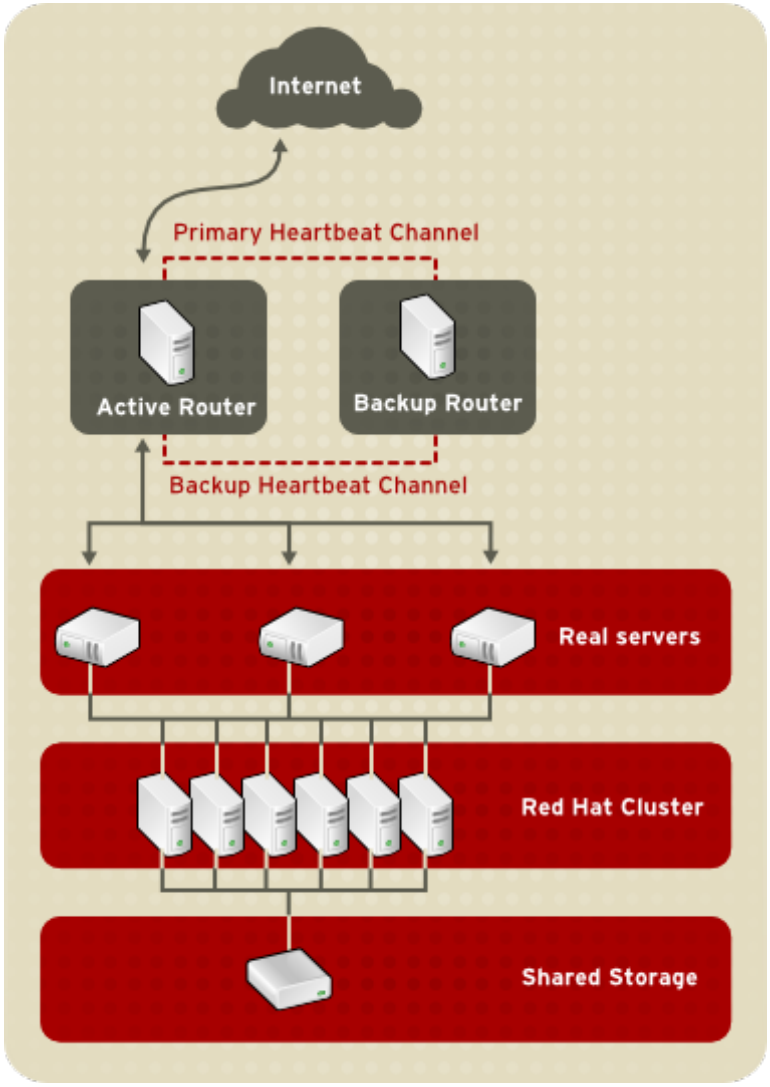

Abbildung 1.2. Eine Three-Tier-Lastverteilungs-Add-On-Konfiguration

Diese Konfiguration eignet sich ideal für gut ausgelastete FTP-Server, bei denen Daten, auf die zugegriffen werden kann, auf einem Hochverfügbarkeits-Server gespeichert werden und auf die von jedem realen Server via einem exportierten NFS-Verzeichnis oder einer Samba-Freigabe zugegriffen wird. Diese Topologie wird außerdem für Websites empfohlen, die auf eine zentrale, Hochverfügbarkeits-Datenbank für Transaktionen zugreifen. Zusätzlich können Administratoren unter Verwendung einer aktiv-aktiv-Konfiguration in Zusammenhang mit Red Hat Cluster Manager ein Hochverfügbarkeits-Cluster so konfigurieren, dass es diese beiden Rollen gleichzeitig ausübt.

Die dritte Tier im oben aufgeführten Beispiel muss Red Hat Cluster Manager nicht nutzen. Wird jedoch keine Hochverfügbarkeitslösung verwendet, stellt dies einen "Single Point of Failure" dar.

1.3. Lastverteilungs-Add-On Scheduling-Überblick

Einer der Vorteile bei der Verwendung des Lastverteilungs-Add-Ons ist die Fähigkeit, flexible Lastverteilung auf IP-Level im Pool der realen Server durchzuführen. Diese Flexibilität ist auf die Vielzahl der Scheduling-Algorithmen zurückzuführen, aus denen ein Administrator bei der Konfiguration des Lastverteilungs-Add-Ons auswählen kann. Lastverteilungs-Add-On-Lastverteilung ist weniger flexiblen Methoden, wie *Round-Robin DNS* überlegen, bei der die hierarchische Art von DNS und das Caching (Zwischenspeichern) von Client-Maschinen zu ungleichmäßiger Auslastung führen kann. Zusätzlich besitzt das Filtern auf einer niedrigen Stufe, das vom LVS-Router eingesetzt wird, Vorteile gegenüber

einer Weiterleitung von Anfragen auf Applikationslevel, da Lastverteilung auf der Stufe eines Netzwerkpakets minimale Rechenleistung verursacht und eine bessere Skalierbarkeit bietet.

Mit Hilfe von Scheduling kann der aktive Router die Aktivität des realen Servers, sowie optional einen vom Administrator zugewiesenen *Gewichtung*s-Faktor beim Routen von Dienstanfragen berücksichtigen. Das Verwenden von zugewiesenen Gewichtungen schafft beliebige Prioritäten für individuelle Maschinen. Wird diese Art von Scheduling verwendet, ist es möglich, eine Gruppe realer Server zu erstellen und eine Kombination aus Hardware und Software zu verwenden. Der aktive Router kann dann die Auslastung gleichmäßig auf jeden realen Server verteilen.

Der Scheduling-Mechanismus für das Lastverteilungs-Add-On wird von einer Sammlung von Kernel-Patches, genannt *IP Virtual Server*- oder *IPVS*-Module, zur Verfügung gestellt. Diese Module aktivieren *Layer 4* (*L4*) Transport-Layer-Switching, welches für eine gute Zusammenarbeit mit mehreren Servern im Rahmen einer einzelnen IP-Adresse konzipiert ist.

Um Pakete effektiv zu den realen Servern zu routen und deren Verlauf nachzuverfolgen, erstellt IPVS eine *IPVS-Tabelle* im Kernel. Diese Tabelle wird vom aktiven LVS-Router verwendet, um Anfragen von einer virtuellen Server-Adresse umzuleiten, die an einen realen Server im Pool gerichtet sind, oder von diesem zurückkommen. Die IPVS-Tabelle wird laufend durch das Dienstprogramm *ipvsadm* aktualisiert — je nach Verfügbarkeit werden Cluster-Mitglieder hinzugefügt oder entfernt.

1.3.1. Scheduling-Algorithmen

Die Struktur, die die IPVS-Tabelle übernimmt, hängt vom Scheduling-Algorithmus ab, den der Administrator für jeden festgelegten virtuellen Server auswählt. Um die maximale Flexibilität hinsichtlich der Diensttypen, die Sie als Cluster zusammenfassen können und bei der Einplanung dieser Dienste zu gewährleisten, liefert Red Hat Enterprise Linux die folgenden, unten aufgeführten Scheduling-Algorithmen. Für Anweisungen zur Zuweisung von Scheduling-Algorithmen, werfen Sie einen Blick auf Abschnitt 4.6.1, „Der Unterabschnitt **VIRTUAL SERVER**“.

Round-Robin-Scheduling

Verteilt jede Anfrage nacheinander in einem Pool von realen Servern. Bei der Verwendung dieses Algorithmus werden alle realen Server als gleichwertig behandelt, ohne Rücksicht auf deren Kapazität oder Auslastung. Dieses Scheduling-Modell ist so ähnlich wie Round-Robin-DNS, ist jedoch tiefgehender, da es netzwerkbasiert und nicht hostbasiert ist. Das Lastverteilungs-Add-On Round-Robin-Scheduling leidet außerdem nicht unter den ungleichmäßigen Auslastungen, die von zwischengespeicherten DNS-Anfragen verursacht werden.

Gewichtetes Round-Robin-Scheduling

Verteilt jede Anfrage nacheinander in einem Pool von realen Servern, teilt jedoch Servern mit einer höheren Kapazität mehr Jobs zu. Die Kapazität wird durch einen durch einen Benutzer zugewiesenen Gewichtungsfaktor angezeigt, der dann durch eine dynamische Auslastungsinformation entweder nach oben oder unten korrigiert wird. Werfen Sie einen Blick auf Abschnitt 1.3.2, „Server-Gewichtung und Scheduling“ für weitere Informationen zur Gewichtung von realen Servern.

Gewichtetes Round-Robin-Scheduling ist die bevorzugte Wahl falls signifikante Unterschiede bei der Kapazität von realen Servern in einem Server-Pool vorliegen. Wenn die Last durch Anfragen jedoch dramatisch variiert, kann ein stark gewichteter Server ggf. mehr Anfragen beantworten, als ihm zugeteilt sind.

Least-Connection

Verteilt mehr Anfragen an reale Server mit weniger aktiven Verbindungen. Da aktive Verbindungen mit den realen Servern mit Hilfe der IPVS-Tabelle nachverfolgt werden, ist Least-Connection eine Form von dynamischem Scheduling-Algorithmus und ist die bessere Wahl, falls ein hohes Maß an Abweichung in der Anfragelast vorliegt. Es ist am besten für einen Pool von realen Server geeignet, in dem jeder Server-Knoten in etwa dieselbe Kapazität besitzt. Falls die realen Server unterschiedliche Kapazitäten besitzen, ist das gewichtete Least-Connection-Scheduling die bessere Wahl.

Gewichtete Least-Connections (Standard)

Verteilt mehrere Anfragen an Server mit weniger aktiven Verbindungen in Relation zu ihrer Kapazität. Die Kapazität wird durch eine benutzerdefinierte Gewichtung angezeigt, welche dann wiederum je nach dynamischer Lastinformation nach oben oder unten korrigiert wird. Der Zusatz der Gewichtung macht diesen Algorithmus zu einer idealen Alternative, wenn der Pool der realen Server aus Hardware mit unterschiedlichen Kapazitäten besteht. Werfen Sie einen Blick auf Abschnitt 1.3.2, „Server-Gewichtung und Scheduling“ für weitere Informationen zur Gewichtung von realen Servern.

Locality-Based Least-Connection Scheduling

Verteilt mehrere Anfragen an Server mit weniger aktiven Verbindungen in Relation zu ihren Ziel-IPs. Dieser Algorithmus ist für den Einsatz in einem Proxy-Cache-Server-Cluster gedacht. Er routet die Pakete für eine IP-Adresse solange an den Server für diese Adresse, bis dieser Server seine Kapazitätsgrenze überschreitet und einen Server mit dessen halber Last vorliegen hat. In diesem Fall weist er die IP-Adresse einem realen Server mit der geringsten Auslastung zu.

Locality-Based Least-Connection Scheduling mit Replication Scheduling

Verteilt mehr Anfragen an Server mit weniger aktiven Verbindungen in Relation zu ihren Ziel-IPs. Dieser Algorithmus ist auch für den Einsatz in einem Proxy-Cache-Server-Cluster gedacht. Es unterscheidet sich vom Locality-Based Least-Connection Scheduling durch das Zuordnen der Ziel-IP-Adresse zu einer Teilmenge von realer Server-Knoten. Anfragen werden anschließend zum Server in dieser Teilmenge mit der niedrigsten Anzahl an Verbindungen geroutet. Falls alle Knoten für die Ziel-IP über der Kapazität liegen, wird ein neuer Server für diese Ziel-IP-Adresse repliziert, indem der reale Server mit den wenigsten Verbindungen aus dem gesamten Pool der realen Server zur Teilmenge der realen Server für diese Ziel-IP hinzugefügt wird. Der Knoten mit der höchsten Last wird dann aus der Teilmenge der realen Server entfernt, um eine Überreplikation zu verhindern.

Destination-Hash-Scheduling

Verteilt Anfragen an den Pool realer Server, indem die Ziel-IP in einer statischen Hash-Tabelle abgefragt wird. Dieser Algorithmus ist für die Verwendung in einem Proxy-Cache Server-Cluster konzipiert.

Source-Hash-Scheduling

Verteilt Anfragen an den Pool realer Server, indem die Quell-IP in einer statischen Hash-Tabelle gesucht wird. Dieser Algorithmus ist für LVS-Router mit mehreren Firewalls konzipiert.

1.3.2. Server-Gewichtung und Scheduling

Der Administrator des Lastverteilungs-Add-Ons kann jedem Knoten in einem Pool von realen Servern

eine *Gewichtung* zuweisen. Diese Gewichtung besteht aus einem ganzzahligen Wert, der bei jedem beliebigen *gewichtungsfähigen* Scheduling-Algorithmus (wie beispielsweise gewichtete Least-Connections) mit einbezogen wird und den LVS-Router dabei unterstützt, Hardware mit unterschiedlichen Fähigkeiten gleichmäßiger auszulasten.

Gewichtungen funktionieren im Verhältnis relativ zueinander. Besitzt ein realer Server beispielsweise eine Gewichtung von 1 und ein anderer Server eine Gewichtung von 5, dann erhält der Server mit der Gewichtung von 5 genau 5 Verbindungen für jede einzelne Verbindung, die der andere Server bekommt. Der Standardwert für eine Gewichtung eines realen Servers ist 1.

Auch wenn das Hinzufügen von Gewichtungen zu unterschiedlichen Hardware-Konfigurationen in einem Pool von realen Servern bei der effektiveren Verteilung der Last des Clusters behilflich sein kann, können temporäre ungleichmäßige Verteilungen bei der Integration eines realen Servers in den Pool der realen Server auftreten, wenn der virtuelle Server für die Verwendung von gewichteten Least-Connections eingeplant ist. Stellen Sie sich beispielsweise vor, es existierten drei Server in einem Pool von realen Servern. Server A und B sind mit 1 gewichtet, und der dritte, Server C ist mit 2 gewichtet. Falls Server C aus irgendwelchen Gründen ausfällt, wird die anfallende zusätzliche Last gleichmäßig zwischen Server A und B verteilt. Sobald Server C jedoch wieder erreichbar ist, ermittelt der LVS-Router, dass er keine Verbindungen besitzt und weist dem Server solange alle eingehenden Anfragen zu, bis er mit Server A und B wieder auf einer Stufe ist.

Um dieses Phänomen zu verhindern, können Administratoren den virtuellen Server zu einem *Quiesce*-Server machen — jedes Mal, wenn ein neuer realer Server-Knoten aktiv wird, wird die Least-Connections-Tabelle wieder auf Null zurückgesetzt und der LVS-Router routet Anfragen so, als ob jeder der realen Server neu zum Cluster hinzugefügt wurde.

1.4. Routing-Methoden

Red Hat Enterprise Linux verwendet *Network Address Translation* oder *NAT-Routing* für das Lastverteilungs-Add-On, was dem Administrator eine hohe Flexibilität beim Einsatz von verfügbarer Hardware und bei der Integration des Lastverteilungs-Add-Ons in ein vorhandenes Netzwerk einräumt.

1.4.1. NAT-Routing

Abbildung 1.3, „Lastverteilungs-Add-On, implementiert mit NAT-Routing“ veranschaulicht das Lastverteilungs-Add-On unter Verwendung von NAT-Routing, um Anfragen zwischen dem Internet- und einem privaten Netzwerk zu verschieben.

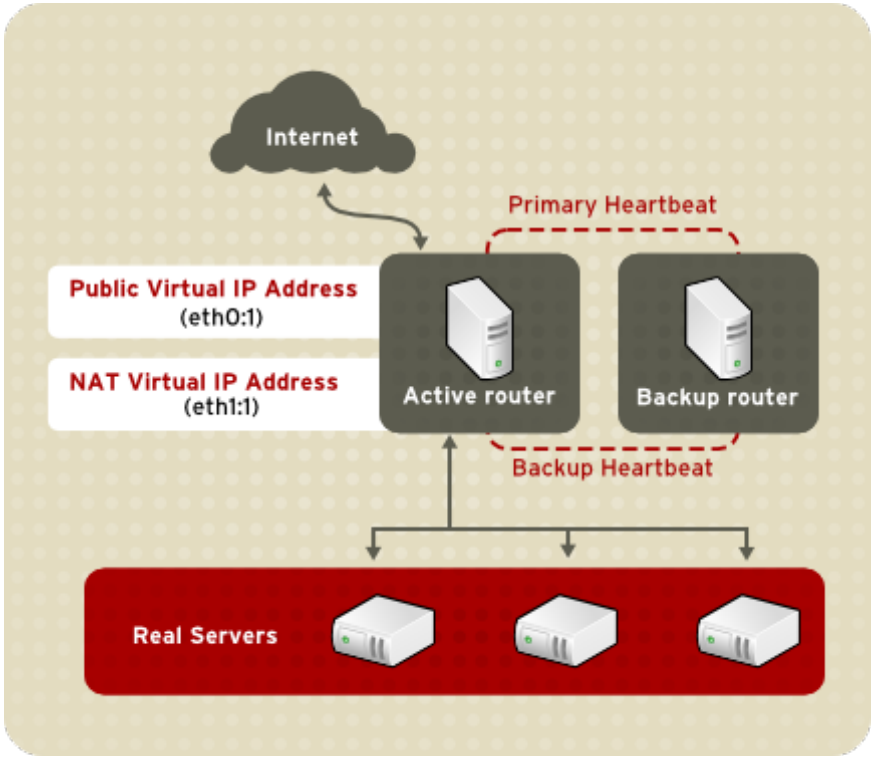

Abbildung 1.3. Lastverteilungs-Add-On, implementiert mit NAT-Routing

In dem Beispiel existieren zwei NICs im aktiven LVS-Router. Die NIC für das Internet besitzt eine *reale IP-Adresse* auf eth0 und eine Floating-IP-Adresse, mit einem Alias zu eth0:1. Die NIC für die private Netzwerkschnittstelle besitzt eine reale IP-Adresse auf eth1 und eine Floating-IP-Adresse, mit einem Alias zu eth1:1. Bei einer Ausfallsicherung werden die virtuelle Schnittstelle in Richtung Internet, und die ins private Netz ausgerichtete virtuelle Schnittstelle gleichzeitig vom Backup-LVS-Router übernommen. Alle realen Server im privaten Netzwerk verwenden die Floating-IP für den NAT-Router als ihre standardmäßige Route, um mit dem aktiven LVS-Router zu kommunizieren, so dass ihre Fähigkeiten, auf Anfragen aus dem Internet zu reagieren, nicht beeinträchtigt werden.

In diesem Beispiel werden für die öffentliche, floating LVS-IP-Adresse des LVS-Router und die private, floating NAT-IP-Adresse zwei Aliase für die physikalischen NICs erstellt. Auch wenn es möglich ist, jede floating IP-Adresse mit ihrem physikalischen Gerät auf den LVS-Router-Knoten zu verknüpfen, ist das Vorhandensein von mehr als zwei NICs keine Voraussetzung.

Bei der Verwendung dieser Topologie erhält der aktive LVS-Router die Anfragen und routet sie an den entsprechenden Server weiter. Der reale Server verarbeitet anschließend die Anfrage und gibt die Pakete an den LVS-Router zurück. Der LVS-Router verwendet Network Address Translation, um die Adresse des realen Servers in den Paketen durch die öffentliche VIP-Adresse der LVS-Router zu ersetzen. Dieser Prozess wird als *IP-Masquerading* bezeichnet, da die tatsächlichen IP-Adressen der realen Server vor den anfragenden Clients versteckt wird.

Mit Hilfe von NAT-Routing können reale Server beliebige Rechner sein, auf denen eine Vielzahl an Betriebssystemen laufen. Der Hauptnachteil von NAT-Routing ist, dass der LVS-Router in größeren Einsatzszenarien zu einer Art Engpass werden kann, da er ausgehende und eingehende Anfragen verarbeiten muss.

1.4.2. Direktes Routing

Der Aufbau einer Lastverteilungs-Add-On-Installation, die direktes Routing verwendet, bietet den Vorteil einer höheren Leistung im Vergleich zu anderen Lastverteilungs-Add-On-Netzwerk-Topologien. Direktes

Routing ermöglicht den realen Servern, Pakete direkt zu verarbeiten und direkt an einen anfragenden Benutzer zu routen, anstatt ausgehende Pakete durch den LVS-Router zu leiten. Direktes Routing vermindert die Wahrscheinlichkeit von Problemen bei der Netzwerkleistung, indem es den Job des LVS-Routers nur auf das Verarbeiten von eingehenden Paketen beschränkt.

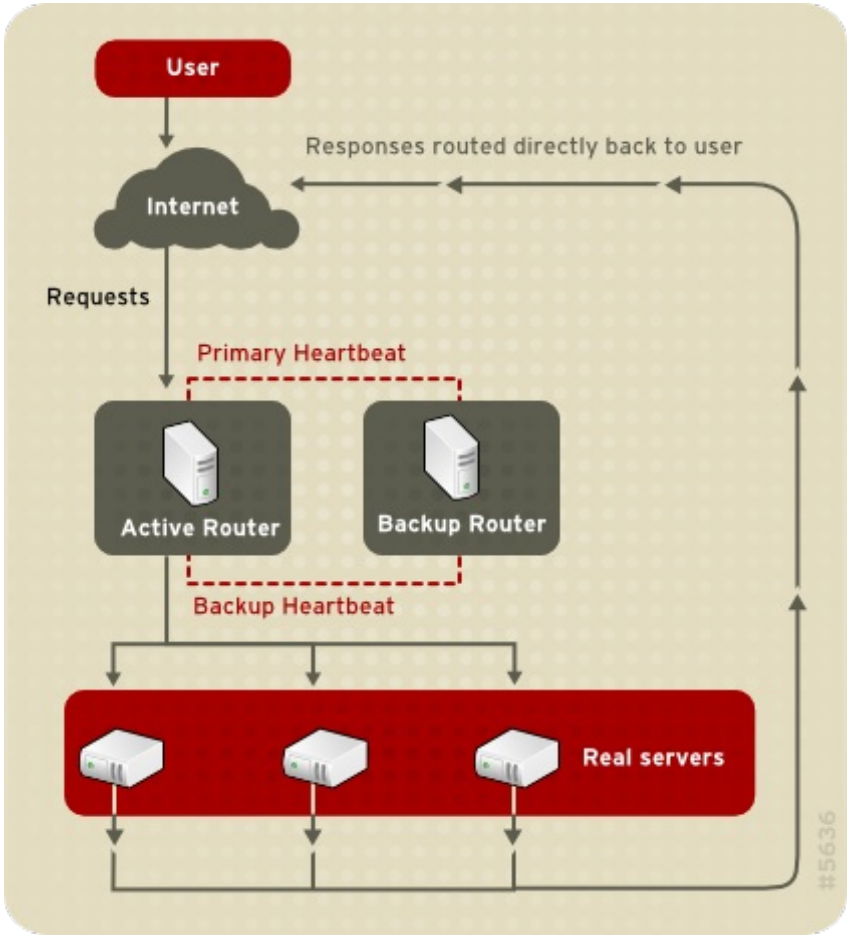

Abbildung 1.4. Lastverteilungs-Add-On, implementiert mit direktem Routing

In einer typischen Direktes-Routing-Lastverteilungs-Add-On-Konfiguration empfängt ein eingehender Server Anfragen durch eine virtuelle IP (VIP) und verwenden einen Plan-Algorithmus, um die Anfragen an reale Server zu routen. Jeder reale Server verarbeitet Anfragen und senden Antworten direkt an Clients und umgeht dabei die LVS-Router. Direktes Routing ermöglicht Skalierbarkeit, so dass reale Server hinzugefügt werden können, ohne dass der LVS-Router zusätzlich belastet wird, ausgehende Pakete vom realen Server zum Client zu routen, was bei hoher Netzwerkauslastung zu einem Engpass führen kann.

1.4.2.1. Direktes Routing und die ARP-Einschränkung

Auch wenn es viele Vorteile zur Verwendung von direktem Routing in Zusammenhang mit dem Lastverteilungs-Add-On gibt, existieren auch einige Einschränkungen. Das häufigste Problem mit direktem Routing und dem Lastverteilungs-Add-On tritt beim *Address Resolution Protocol* (ARP) auf.

In typischen Situationen sendet ein Client aus dem Internet eine Anfrage an eine IP-Adresse. Netzwerk-Router senden Anfragen typischerweise an ihr Ziel, indem sie IP-Adressen mit ARP mit einer MAC-Adresse einer Maschine verbinden. ARP-Anfragen werden an alle im Netzwerk verbundenen Maschinen verbreitet und die Maschine mit der korrekten IP-/MAC-Adresskombination erhält das Paket. Die IP-/MAC-Verbindungen werden in einem ARP-Cache gespeichert, welcher periodisch gelöscht (normalerweise alle 15 Minuten) und neu mit IP-/MAC-Verbindungen gefüllt wird.

Das Problem mit ARP-Anfragen in einer Direkt-Routing Lastverteilungs-Add-On-Konfiguration ist, dass aufgrund der Tatsache, dass eine Client-Anfrage an eine IP-Adresse mit einer MAC-Adresse verknüpft sein muss, damit diese Anfrage behandelt werden kann, die virtuelle IP-Adresse des Lastverteilungs-Add-On-Systems ebenfalls mit einer MAC verknüpft sein muss. Da aber sowohl der LVS-Router, als auch die realen Server dieselbe VIP besitzen, wird die ARP-Anfrage an alle mit dem VIP verknüpften Knoten verbreitet. Dies kann zu einigen Problemen führen, z.B. dass die VIP direkt mit einem der realen Server verknüpft wird und Anfragen direkt weiterleitet und dabei den LVS-Router komplett umgeht, was den Zweck der Lastverteilungs-Add-On-Konfiguration außer Kraft setzt.

Um dieses Problem zu lösen, sollten Sie sicherstellen, dass eingehende Anfragen immer an den LVS-Router, anstatt an einen der realen Server gesendet werden. Dies erreichen Sie entweder mit Hilfe von **`arptables_jf`** oder des Paketfilter-Tools **`iptables`** aus den folgenden Gründen:

- **`arptables_jf`** hindert ARP an der Verknüpfung von VIPs mit realen Servern.
- Die **`iptables`**-Methode umgeht das ARP-Problem komplett, indem VIPs auf realen Servern von vornherein nicht konfiguriert werden.

Werfen Sie einen Blick auf Abschnitt 3.2.1, „Direktes Routing und **`arptables_jf`**“ oder Abschnitt 3.2.2, „Direktes Routing und **`iptables`**“ für weitere Informationen zur Verwendung von **`arptables`** oder **`iptables`** in einer Lastverteilungs-Add-On-Umgebung mit direktem Routing.

1.5. Persistenz und Firewall-Markierungen

In bestimmten Situationen kann es wünschenswert sein, dass sich ein Client immer wieder mit demselben realen Server erneut verbindet, anstatt dass ein Lastverteilungs-Add-On Lastverteilungs-Algorithmus diese Anfrage an den am besten verfügbaren Server zur Lastverteilung schickt. Beispiele für eine solche Situation umfassen Web-Formulare, die sich über mehrere Bildschirme erstrecken, Cookies, SSL- und FTP-Verbindungen. In diesen Fällen funktioniert ein Client ggf. nicht ordnungsgemäß, bis die Transaktionen von demselben Server behandelt werden, um den Kontext beizubehalten. Das Lastverteilungs-Add-On bietet zwei verschiedene Features, um dies zu handhaben: *Persistenz* und *Firewall Markierungen*.

1.5.1. Persistenz

Sofern aktiviert, agiert die Persistenz wie ein Timer. Wenn sich ein Client mit einem Dienst verbindet, merkt sich das Lastverteilungs-Add-On die letzte Verbindung für eine angegebene Zeitspanne. Falls sich dieselbe Client-IP-Adresse erneut innerhalb dieser Spanne verbindet, wird es an denselben Server weitergeleitet, mit dem sie sich zuvor bereits verbunden hatte — dabei werden die Mechanismen zur Lastverteilung übergangen. Falls eine Verbindung außerhalb des Zeitfensters zustandekommt, wird sie anhand der Scheduling-Regeln, die gerade in Kraft sind, behandelt.

Persistenz erlaubt dem Administrator auch die Angabe einer Subnet-Maske, die für den Test der Client-IP-Adresse angewendet werden kann, als ein Tool zur Kontrolle, welche Adressen ein höheres Level an Persistenz besitzen, so dass Verbindungen mit diesem Subnet gruppiert werden.

Das Gruppieren von Verbindungen, die für verschiedene Ports bestimmt sind, kann für Protokolle von Bedeutung sein, die mehr als einen Port für die Kommunikation verwenden, wie beispielsweise FTP. Allerdings stellt Persistenz nicht die effektivste Art und Weise dar, Probleme mit der Gruppierung von Verbindungen, die für verschiedene Ports bestimmt sind, zu behandeln. Für diese Situationen werden am besten *Firewall-Markierungen* vewendet.

1.5.2. Firewall-Markierungen

Firewall-Markierungen sind eine leichte und effektive Art und Weise, um Ports zu gruppieren, die für ein

Protokoll oder eine Gruppe von verwandten Protokollen verwendet werden. Falls das Lastverteilungs-Add-On beispielsweise im Rahmen einer E-Commerce-Site eingesetzt wird, können Firewall-Markierungen dazu verwendet werden, HTTP-Verbindungen auf Port 80, sowie sichere HTTPS-Verbindungen auf Port 443 zu bündeln. Indem den virtuellen Servern dieselbe Firewall-Markierung für jedes Protokoll zugewiesen wird, können Informationen über den Zustand der Transaktion erhalten bleiben, da der LVS-Router alle Anfragen an denselben realen Server weiterleitet, nachdem eine Verbindung geöffnet wurde.

Aufgrund seiner Effizienz und der einfachen Handhabung sollten Administratoren des Lastverteilungs-Add-On wann immer möglich Firewall-Markierungen zugunsten von Persistenz für Gruppierungsverbindungen verwenden. Allerdings sollten Administratoren noch Persistenz in Verbindung mit Firewall-Markierungen zu den virtuellen Servern hinzufügen um sicherzustellen, dass Clients wieder mit demselben Server für eine angemessene Zeitspanne verbunden werden.

1.6. Lastverteilungs-Add-On — ein Block-Diagramm

LVS-Router verwenden eine Sammlung von Programmen, um Cluster-Mitglieder und Cluster-Dienste zu überwachen. Abbildung 1.5, „Komponenten des Lastverteilungs-Add-Ons“ veranschaulicht, wie diese verschiedenen Programme sowohl auf den aktiven, als auch auf den Backup LVS-Routern zusammenarbeiten, um das Cluster zu verwalten.

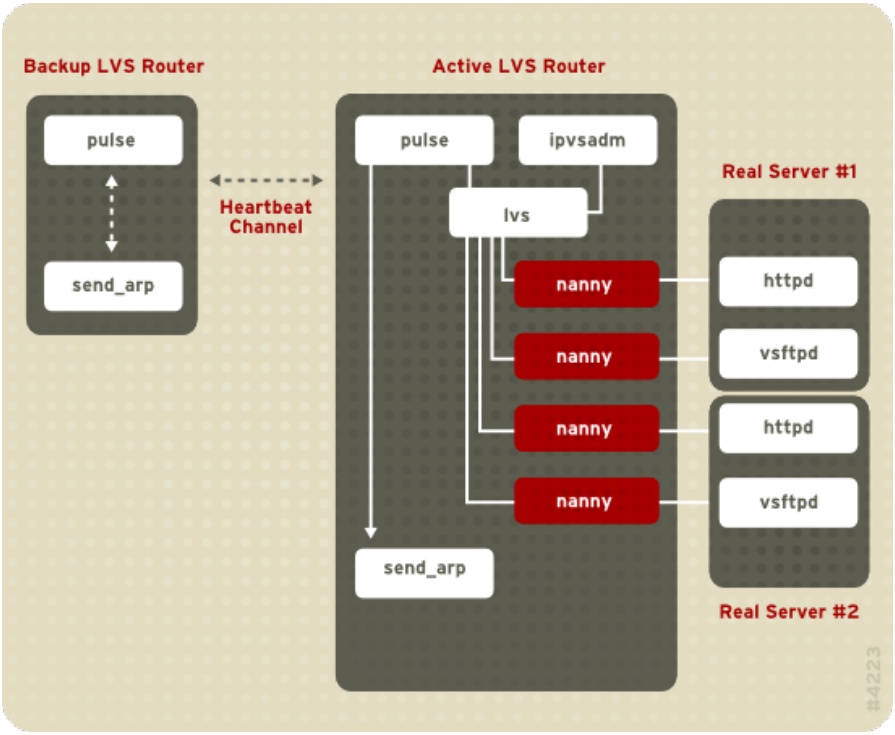

Abbildung 1.5. Komponenten des Lastverteilungs-Add-Ons

Der **pulse**-Daemon läuft sowohl auf den aktiven, als auch auf den passiven LVS-Routern. Auf dem Backup-LVS-Router sendet **pulse** einen *heartbeat* an die öffentliche Schnittstelle des aktiven Routers, um sicherzustellen, dass der aktive LVS-Router ordnungsgemäß funktioniert. Auf dem aktiven LVS-Router startet **pulse** den **lvs**-Daemon und reagiert auf *heartbeat*-Anfragen des Backup-LVS-Routers.

Sobald er gestartet wird, ruft der **lvs**-Daemon das **ipvsadm**-Dienstprogramm auf, um die IPVS (IP Virtual Server) Routing-Tabelle im Kernel zu konfigurieren und zu pflegen, und startet einen **nanny**-Prozess für jeden konfigurierten virtuellen Server auf jedem realen Server. Jeder **nanny**-Prozess

überprüft den Status eines konfigurierten Dienstes auf einem realen Server und informiert den **lvs**-Daemon, falls der Dienst auf diesem realen Server nicht korrekt funktioniert. Falls eine Fehlfunktion entdeckt wird, weist der **lvs**-Daemon **ipvsadm** an, diesen realen Server aus der IPVS-Routing-Tabelle zu entfernen.

Falls der Backup-LVS-Router keine Antwort vom aktiven LVS-Router erhält, leitet er eine Ausfallsicherung ein, indem er **send_arp** aufruft, alle virtuellen IP-Adressen den NIC Hardware-Adressen (*MAC* Adressen) des Backup-LVS-Router neu zuzuweisen. Weiterhin sendet er einen Befehl an den aktiven LVS-Router, sowohl durch die öffentliche, als auch die private Netzwerkschnittstelle, den **lvs**-Daemon auf dem aktiven LVS-Router zu beenden, und startet den **lvs**-Daemon auf dem Backup-LVS-Router, um Anfragen für die konfigurierten virtuellen Server zu akzeptieren.

1.6.1. Komponenten des Lastverteilungs-Add-Ons

Abschnitt 1.6.1.1, „**pulse**“ zeigt eine detaillierte Liste von jeder Software-Komponente in einem LVS-Router an.

1.6.1.1. pulse

Dies ist der Kontrollprozess, der alle anderen Daemons startet, die mit den LVS-Routern in Verbindung stehen. Zum Zeitpunkt des Bootens wird der Daemon durch das Skript **/etc/rc.d/init.d/pulse** gestartet und liest anschließend die Konfigurationsdatei **/etc/sysconfig/ha/lvs.cf**. Auf dem aktiven Router startet **pulse** den LVS-Daemon. Auf dem Backup-Router ermittelt **pulse** den Zustand des aktiven Routers durch die Ausführung eines einfachen Heartbeats zu einem vom Benutzer konfigurierbaren Intervall. Falls der aktive Router nach einem vom Benutzer konfigurierbaren Intervall nicht antwortet, wird die Ausfallsicherung eingeleitet. Während der Ausfallsicherung weist **pulse** auf dem Backup-Router den **pulse**-Daemon auf dem aktiven Router an, alle LVS-Dienste zu beenden, startet das **send_arp**-Programm, um die floating IP-Adressen erneut der MAC-Adresse des Backup-Routers zuzuweisen und startet den **lvs**-Daemon.

1.6.1.2. lvs

Der **lvs**-Daemon läuft auf dem aktiven LVS-Router, sobald er von **pulse** aufgerufen wird. Er liest die Konfigurationsdatei **/etc/sysconfig/ha/lvs.cf**, ruft das Dienstprogramm **ipvsadm** auf, um die IPVS-Routing-Tabelle zu erstellen und zu pflegen und weist jedem konfigurierten Lastverteilungs-Add-On-Dienst einen **nanny**-Prozess zu. Falls **nanny** meldet, dass ein realer Server nicht mehr erreichbar ist, weist **lvs** das Dienstprogramm **ipvsadm** an, den realen Server aus der IPVS-Routing-Tabelle zu entfernen.

1.6.1.3. ipvsadm

Dieser Dienst aktualisiert die IPVS-Routing-Tabelle im Kernel. Der **lvs**-Daemon richtet das Lastverteilungs-Add-On ein und verwaltet dieses, indem er **ipvsadm** aufruft, um Einträge in der IPVS-Routing-Tabelle hinzuzufügen, zu verändern oder zu löschen.

1.6.1.4. nanny

Der **nanny**-Überwachungs-Daemon läuft auf dem aktiven LVS-Router. Mit Hilfe dieses Daemons ermittelt der aktive Router den Zustand jedes realen Servers und überwacht optional dessen Arbeitsbelastung. Ein separater Prozess läuft für jeden Dienst, der auf jedem realen Server definiert ist.

1.6.1.5. /etc/sysconfig/ha/lvs.cf

Dies ist die Lastverteilungs-Add-On-Konfigurationsdatei. Alle Daemons erhalten ihre Konfigurationsinformationen direkt oder indirekt aus dieser Datei.

1.6.1.6. Piranha Configuration Tool

Dies ist das webbasierte Tool für die Überwachung, die Konfiguration und die Administration des Lastverteilungs-Add-On. Es ist das Standard-Tool zur Pflege der Lastverteilungs-Add-On-Konfigurationsdatei **/etc/sysconfig/ha/lvs.cf**.

1.6.1.7. send_arp

Dieses Programm sendet während der Ausfallsicherung ARP-Broadcasts beim Übertragen von IP-Adressänderungen von einem Knoten auf einen anderen.

Kapitel 2, *Erstmalige Konfiguration des Lastverteilungs-Add-Ons* prüft wichtige Konfigurationsschritte nach der Installation, die Sie vor der Konfiguration von Red Hat Enterprise Linux als LVS-Router durchführen sollten.

---

[1] Ein virtueller Server ist ein Dienst, der so konfiguriert ist, dass er auf einer speziellen virtuellen IP horcht. Werfen Sie einen Blick auf Abschnitt 4.6, „**VIRTUAL SERVERS**“ für weitere Informationen zur Konfiguration eines virtuellen Servers unter Verwendung des **Piranha Configuration Tool**.

Kapitel 2. Erstmalige Konfiguration des Lastverteilungs-Add-Ons

Nach der Installation von Red Hat Enterprise Linux müssen Sie einige grundlegende Schritte durchführen, um sowohl die LVS-Router, als auch die realen Server einzurichten. Dieses Kapitel deckt die ersten Schritte hierfür im Detail ab.

Anmerkung

Der LVS-Router-Knoten, der zum aktiven Knoten wird, sobald das Lastverteilungs-Add-On gestartet wird, wird auch als der *Primäre Knoten* bezeichnet. Verwenden Sie zur Konfiguration des Lastverteilungs-Add-On **Piranha Configuration Tool** auf dem primären Knoten.

2.1. Konfiguration von Diensten auf den LVS-Routern

Das Installationsprogramm von Red Hat Enterprise Linux installiert alle Komponenten, die zur Einrichtung des Lastverteilungs-Add-Ons benötigt werden. Die entsprechenden Dienste müssen jedoch vor der Konfiguration des Lastverteilungs-Add-Ons aktiviert werden. Richten Sie die entsprechenden Dienste für beide LVS-Router so ein, dass sie zum Zeitpunkt des Bootens gestartet werden. Es stehen drei Haupt-Tools zur Einrichtung von Diensten unter Red Hat Enterprise Linux zur Verfügung, so dass diese zum Zeitpunkt des Bootens aktiviert werden: Das Kommandozeilenprogramm **chkconfig**, das auf Ncurses basierende Programm **ntsysv** und das grafische **Services Configuration Tool**. Alle drei Tools erfordern Root-Zugriff.

Anmerkung

Um Root-Zugriff zu erlangen, öffnen Sie einen Shell-Prompt und verwenden den Befehl **su -**, gefolgt vom Root-Passwort. Zum Beispiel:

```
$ su - Root-Passwort
```

Auf den LVS-Routern müssen drei Dienste zum Zeitpunkt des Bootens auf "aktiviert" gesetzt werden:

- Der Dienst **piranha-gui** (nur primärer Knoten)
- Der **pulse**-Dienst
- Der **sshd**-Dienst

Falls Sie Mehrport-Dienste zu einem Cluster zusammenfassen oder Firewall-Markierungen verwenden, müssen Sie außerdem den **iptables**-Dienst aktivieren.

Am besten werden diese Dienste so eingerichtet, dass sie sowohl in Runlevel 3, als auch in Runlevel 5 aktiviert werden. Geben Sie den folgenden Befehl für jeden Dienst ein, um dies mit **chkconfig** zu erledigen:

/sbin/chkconfig --level 35 *daemon* on

Ersetzen Sie im oben aufgeführten Befehl ***daemon*** mit dem Namen des Dienstes, den Sie aktivieren. Um eine Liste von Diensten auf dem System zu bekommen, und darüber hinaus zu sehen, für welche Runlevel diese aktiviert sind, führen Sie den folgenden Befehl aus:

```
/sbin/chkconfig --list
```

Warnung

Das Aktivieren eines der oben aufgeführten Dienste unter Verwendung von **chkconfig** aktiviert den Daemon nicht wirklich. Dazu müssen Sie den Befehl **/sbin/service** verwenden. Werfen Sie einen Blick auf Abschnitt 2.3, „Starten des **Piranha Configuration Tool** Dienstes“ für ein Beispiel zur Verwendung des **/sbin/service**-Befehls.

Für weitere Informationen zu Runlevels und zur Konfiguration von Diensten mit **ntsysv** und dem **Services Configuration Tool** werfen Sie einen Blick auf das Kapitel mit dem Titel *"Zugriffskontrolle auf Dienste"* im *Red Hat Enterprise Linux System Administration Guide*.

2.2. Einrichten eines Passworts für das Piranha Configuration Tool

Bevor das **Piranha Configuration Tool** erstmalig auf dem primären LVS-Router verwendet werden kann, müssen Sie den Zugriff auf dasselbe mit einem Passwort einschränken. Loggen Sie sich hierfür als Root ein und führen Sie den folgenden Befehl aus:

```
/usr/sbin/piranha-passwd
```

Erstellen Sie nach der Eingabe dieses Befehls das administrative Passwort, wenn Sie dazu aufgefordert werden.

Warnung

Damit ein Passwort sicherer ist, sollte es keine echten Substantive, häufig verwendete Akronyme oder Wörter aus einem Wörterbuch einer beliebigen Sprache enthalten. Hinterlassen Sie das Passwort nicht unverschlüsselt irgendwo auf dem System.

Falls sich das Passwort während einer aktiven Sitzung des **Piranha Configuration Tool** ändert, wird der Administrator aufgefordert, das neue Passwort einzugeben.

2.3. Starten des Piranha Configuration Tool Dienstes

Nachdem Sie das Passwort für das **Piranha Configuration Tool** gesetzt haben, starten Sie den **piranha-gui**-Dienst, der sich unter **/etc/rc.d/init.d/piranha-gui** befindet (neu). Führen Sie hierfür den folgenden Befehl als Root aus:

```
/sbin/service piranha-gui start
```

oder

```
/sbin/service piranha-gui restart
```

Das Ausführen dieses Befehls startet eine private Sitzung des Apache HTTP Server durch den Aufruf der symbolischen Verknüpfung **/usr/sbin/piranha_gui -> /usr/sbin/httpd**. Aus Sicherheitsgründen läuft die **piranha-gui**-Version des **httpd** als Benutzer "piranha" in einem separaten Prozess. Die Tatsache, dass **piranha-gui** den **httpd**-Dienst beeinflusst, bedeutet

Folgendes:

1. Der Apache HTTP Server muss auf dem System installiert sein.
2. Das Anhalten oder Neustarten des Apache HTTP Server durch den Befehl **service** hält den **piranha-gui**-Dienst an.

Warnung

Falls der Befehl **/sbin/service httpd stop** oder **/sbin/service httpd restart** auf einem LVS-Router ausgeführt wird, müssen Sie den **piranha-gui**-Dienst durch die Eingabe des folgenden Befehls starten:
/sbin/service piranha-gui start

Der **piranha-gui**-Dienst ist alles, was benötigt wird, um mit der Konfiguration des Lastverteilungs-Add-Ons zu beginnen. Falls Sie jedoch das Cluster von Remote aus konfigurieren, wird außerdem der **sshd**-Dienst benötigt. Sie müssen den **pulse**-Dienst *nicht* vor Abschluss der Konfiguration unter Verwendung des **Piranha Configuration Tool** starten. Werfen Sie einen Blick auf Abschnitt 4.8, „Das Lastverteilungs-Add-On starten“ für Informationen zum Starten des **pulse**-Dienstes.

2.3.1. Konfiguration des Web-Server-Ports für das Piranha Configuration Tool

Das **Piranha Configuration Tool** läuft standardmäßig auf Port 3636. Um diese Portnummer zu ändern, ändern Sie die Zeile **Listen 3636** im zweiten Abschnitt der **piranha-gui** Web-Server Konfigurationsdatei **/etc/sysconfig/ha/conf/httpd.conf**.

Um das **Piranha Configuration Tool** zu nutzen, benötigen Sie mindestens einen text-basierten Web-Browser. Falls Sie einen Web-Browser auf dem primären LVS-Router starten, öffnen Sie die URL **http://*localhost*:3636**. Sie erreichen das **Piranha Configuration Tool** von überall via Web-Browser, indem Sie ***localhost*** mit dem Hostnamen oder der IP-Adresse des primären LVS-Routers ersetzen.

Wenn sich Ihr Browser mit dem **Piranha Configuration Tool** verbindet, müssen Sie sich für den Zugriff auf die Konfigurationsdienste einloggen. Geben Sie **piranha** im Feld **Username** und das mit **piranha-passwd** erstellte Passwort im Feld **Password** ein.

Sobald das **Piranha Configuration Tool** läuft, sollten Sie ggf. in Erwägung ziehen, den Zugriff auf dieses Tool via Netzwerk einzuschränken. Der nächste Abschnitt beschäftigt sich mit einigen Varianten zur Umsetzung dieser Aufgabe.

2.4. Zugangsbeschränkung zum Piranha Configuration Tool

Das **Piranha Configuration Tool** fordert Sie zur Eingabe eines gültigen Benutzernamens und eines Passworts auf. Da jedoch jegliche an das **Piranha Configuration Tool** gesendete Daten in reiner Textform vorliegen, wird empfohlen, dass Sie den Zugriff nur auf vertrauenswürdige Netzwerke oder die lokale Maschine beschränken.

Die einfachste Variante zur Zugriffsbeschränkung ist die Verwendung der integrierten Zugriffskontrollmechanismen von Apache HTTP Server durch die Bearbeitung von **/etc/sysconfig/ha/web/secure/.htaccess**. Nach der Änderung dieser Datei müssen Sie den **piranha-gui**-Dienst nicht neu starten, da der Server die Datei **.htaccess** bei jedem Zugriff auf das Verzeichnis überprüft.

Standardmäßig gestattet die Zugriffskontrolle für dieses Verzeichnis jedem Benutzer die Anzeige des Inhalts des Verzeichnisses. So sieht der standardmäßige Zugriff aus:

```
Order deny,allow
Allow from all
```

Um den Zugriff auf das **Piranha Configuration Tool** auf 'localhost' zu beschränken, ändern Sie die Datei **.htaccess** so, dass der Zugriff nur noch vom Loopback-Gerät (127.0.0.1) aus erlaubt ist. Werfen Sie einen Blick auf das Kapitel mit dem Titel *Netzwerkskripte* im *Red Hat Enterprise Linux Reference Guide* für weitere Informationen zum Loopback-Gerät.

```
Order deny,allow
Deny from all
Allow from 127.0.0.1
```

Sie können auch spezielle Hosts oder Subnetze autorisieren, wie in diesem Beispiel gezeigt:

```
Order deny,allow
Deny from all
Allow from 192.168.1.100
Allow from 172.16.57
```

In diesem Beispiel können nur Web-Browser von den Maschinen mit der IP-Adresse 192.168.1.100 und von Maschinen aus dem 172.16.57/24 Netzwerk auf das **Piranha Configuration Tool** zugreifen.

Warnung

Das Bearbeiten der **.htaccess**-Datei des **Piranha Configuration Tool** schränkt den Zugriff auf die Konfigurationsseiten im Verzeichnis **/etc/sysconfig/ha/web/secure/** ein, jedoch nicht für die Login- und Hilfeseiten in **/etc/sysconfig/ha/web/**. Um den Zugriff auf dieses Verzeichnis einzuschränken, müssen Sie eine **.htaccess**-Datei im Verzeichnis **/etc/sysconfig/ha/web/** erstellen, und zwar mit zu **/etc/sysconfig/ha/web/secure/.htaccess** identischen Zeilen **order**, **allow**, und **deny**.

2.5. Aktivierung der Paketweiterleitung

Damit der LVS-Router Netzwerkpakete ordnungsgemäß an den realen Server weiterleiten kann, muss auf jedem LVS-Router IP-Weiterleitung im Kernel aktiviert sein. Loggen Sie sich als Root ein und ändern Sie die Zeile **net.ipv4.ip_forward = 0** in **/etc/sysctl.conf** wie folgt:

```
net.ipv4.ip_forward = 1
```

Die Änderungen werden bei einem Neustart des Systems wirksam.

Um zu prüfen, ob IP-Weiterleitung aktiviert ist, geben Sie den folgenden Befehl als Root ein:

```
/sbin/sysctl net.ipv4.ip_forward
```

Falls der oben aufgeführte Befehl eine **1** zurückgibt, dann ist die IP-Weiterleitung aktiviert. Falls der Rückgabewert **0** lautet, dann können Sie die IP-Weiterleitung manuell mit dem folgenden Befehl aktivieren:

```
/sbin/sysctl -w net.ipv4.ip_forward=1
```

2.6. Konfiguration von Diensten auf den realen Servern

Falls die realen Server Red Hat Enterprise Linux-Systeme sind, richten Sie die entsprechenden Server-Daemons so ein, dass sie zum Zeitpunkt des Bootens aktiviert werden. Diese Daemons können **`httpd`** für Web-Dienste oder **`xinetd`** für FTP- oder Telnet-Dienste umfassen.

Es kann ggf. auch nützlich sein, von remote aus auf die realen Server zuzugreifen, so dass der **sshd**-Daemon auch installiert sein und laufen sollte.

Kapitel 3. Einrichten des Lastverteilungs-Add-Ons

Das Lastverteilungs-Add-On besteht aus zwei grundlegenden Gruppen: den LVS-Routern und den realen Servern. Um einen "Single Point of Failure" auszuschließen, sollte jede Gruppe mindestens zwei Mitgliedersysteme enthalten.

Die LVS-Router-Gruppe sollte aus zwei identischen oder sehr ähnlichen Systemen bestehen, auf denen Red Hat Enterprise Linux läuft. Einer agiert als der aktive LVS-Router, während der andere für den unmittelbaren Einsatz bereitsteht, so dass beide so weit wie möglich dieselben Fähigkeiten besitzen müssen.

Vor der Auswahl und der Konfiguration der Hardware für die Gruppe der realen Server müssen Sie ermitteln, welcher der drei Typen der Lastverteilungs-Add-On-Topologien verwendet werden soll.

3.1. Das NAT Lastverteilungs-Add-On-Netzwerk

Die NAT-Topologie bietet einen großen Spielraum bei der Verwendung von existierender Hardware, ist jedoch bei der Handhabung von großer Auslastung in seiner Fähigkeit auf Grund der Tatsache eingeschränkt, dass alle Pakete, die in den Pool ein-, bzw. austreten, den Lastverteilungs-Add-On-Router passieren müssen.

Netzwerk-Layout

Die Topologie für das Lastverteilungs-Add-On unter Verwendung von NAT-Routing ist aus der Sicht eines Netzwerk-Layouts die einfachste, da nur ein Zugriffspunkt zum öffentlichen Netzwerk benötigt wird. Die realen Server geben alle Anfragen zurück an den LVS-Router, so dass sie sich in ihrem eigenen privaten Netzwerk befinden.

Hardware

Die NAT-Topologie ist die flexibelste hinsichtlich der Hardware, weil die realen Server keine Linux-Maschinen sein müssen, um ordnungsgemäß zu funktionieren. In einer NAT-Topologie benötigt jeder reale Server nur eine NIC, da er lediglich dem LVS-Router antwortet. Die LVS-Router dagegen benötigen jeweils zwei NICs, um den Datenverkehr zwischen den beiden Netzwerken zu routen. Da diese Topologie zu einem Engpass beim LVS-Router führt, können auf jedem LVS-Router Gigabit Ethernet-NICs eingesetzt werden, um die Bandbreite, die die LVS-Router handhaben können, zu erweitern. Falls ein Gigabit-Ethernet auf den LVS-Routern eingesetzt wird, muss jedes Switch, das die realen Server mit den LVS-Routern verbindet, mindestens zwei Gigabit-Ethernet-Ports besitzen, um die Auslastung effektiv zu bewältigen.

Software

Da die NAT-Topologie die Verwendung von **`iptables`** für einige Konfigurationen erfordert, kann einiges an Software-Konfiguration über **Piranha Configuration Tool** hinaus anfallen. Speziell FTP-Dienste und die Verwendung von Firewall-Markierungen erfordern zusätzliche manuelle Konfiguration der LVS-Router, um Anfragen ordnungsgemäß zu routen.

3.1.1. Konfiguration von Netzwerkschnittstellen für das Lastverteilungs-Add-On mit NAT

Um das Lastverteilungs-Add-On mit NAT einzurichten, muss der Administrator zunächst die Schnittstellen für das öffentliche und private Netzwerk auf den LVS-Routern konfigurieren. In diesem Beispiel befinden sich die öffentlichen Schnittstellen (**`eth0`**) der LVS-Router im 192.168.26/24 Netzwerk (Ich weiß, Ich weiß, dies ist keine routbare IP, aber gehen wir mal davon aus, dass zusätzlich eine

Firewall vor dem LVS-Router existiert) und die private Schnittstellen, die die Verbindung mit den realen Servern herstellen (**eth1**), im Netzwerk 10.11.12/24.

Demnach könnte auf dem aktiven oder *primären* LVS-Router-Knoten das Netzwerkskript **/etc/sysconfig/network-scripts/ifcfg-eth0** auf der öffentlichen Schnittstelle ungefähr so aussehen:

```
DEVICE=eth0
BOOTPROTO=static
ONBOOT=yes
IPADDR=192.168.26.9
NETMASK=255.255.255.0
GATEWAY=192.168.26.254
```

Das Skript **/etc/sysconfig/network-scripts/ifcfg-eth1** für die private NAT-Schnittstelle auf dem LVS-Router könnte ungefähr so aussehen:

```
DEVICE=eth1
BOOTPROTO=static
ONBOOT=yes
IPADDR=10.11.12.9
NETMASK=255.255.255.0
```

In diesem Beispiel ist die VIP für die öffentliche Schnittstelle des LVS-Routers 192.168.26.10 und die VIP für die NAT- oder private Schnittstelle 10.11.12.10. Es ist daher essentiell, dass die realen Server Anfragen zurück an das VIP für die NAT-Schnittstelle zurückrouten.

Wichtig

Die beispielhaften Konfigurationseinstellungen in diesem Abschnitt für die Ethernet-Schnittstelle gelten für die realen IP-Adressen eines LVS-Routers und *nicht* für die floating IP-Adressen. Um die öffentlichen und privaten floating IP-Adressen zu konfigurieren, sollte der Administrator das **Piranha Configuration Tool** verwenden, wie unter Abschnitt 4.4, „GLOBAL SETTINGS“ und Abschnitt 4.6.1, „Der Unterabschnitt VIRTUAL SERVER“ aufgezeigt.

Konfigurieren Sie nach der Konfiguration der Netzwerkschnittstellen des primären LVS-Router-Knotens die realen Netzwerkschnittstellen des Backup-LVS-Routers — Achten Sie dabei darauf, dass keine der IP-Adressen mit anderen IP-Adressen im Netzwerk kollidieren.

Wichtig

Stellen Sie sicher, dass jede Schnittstelle auf dem Backup-Knoten dasselbe Netzwerk wie die Schnittstelle auf dem primären Knoten bedient. Wenn eth0 beispielsweise mit dem öffentlichen Netzwerk auf dem primären Knoten verbunden ist, muss es ebenfalls mit dem öffentlichen Netzwerk auf dem Backup-Knoten verbunden sein.

3.1.2. Routing auf den realen Servern

Das Wichtigste, das bei der Konfiguration der Netzwerkschnittstellen der realen Server in einer NAT-Topologie beachtet werden muss, ist das Einstellen eines Gateways für die NAT Floating-IP-Adressen des LVS-Routers. In diesem Beispiel lautet die Adresse 10.11.12.10.

Anmerkung

Sobald die Netzwerkschnittstellen auf den realen Servern aktiv sind, können die Maschinen nicht mehr pingen, oder sich auf andere Weise mit dem öffentlichen Netzwerk verbinden. Dies ist normal. Sie sind jedoch in der Lage, die reale IP für die private Schnittstelle des LVS Routers anzupingen, in diesem Fall 10.11.12.8.

Die Datei **/etc/sysconfig/network-scripts/ifcfg-eth0** des realen Servers könnte demnach ähnlich wie folgt aussehen:

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=static
IPADDR=10.11.12.1
NETMASK=255.255.255.0
GATEWAY=10.11.12.10
```

Warnung

Falls ein realer Server mehr als eine Netzwerkschnittstelle mit einer Zeile **GATEWAY=** konfiguriert hat, erhält die erste, die aktiv ist, die Gateway-Adresse. Wenn daher sowohl **eth0** und **eth1** konfiguriert sind und **eth1** für das Lastverteilungs-Add-On verwendet wird, kann es sein, dass die realen Server Anfragen ggf. nicht korrekt routen.
Irrelevante Netzwerkschnittstellen sollten am besten deaktiviert werden, indem ihre Netzwerkskripte im Verzeichnis **/etc/sysconfig/network-scripts/** auf **ONBOOT=no** gesetzt werden oder indem sichergestellt wird, dass das Gateway bei der Schnittstelle korrekt eingerichtet wurde, die als erstes aktiviert wird.

3.1.3. Aktivierung von NAT-Routing auf den LVS-Routern

In einer einfachen NAT Lastverteilungs-Add-On-Konfiguration, in dem jeder geclusterte Dienst nur einen Port verwendet, wie HTTP auf Port 80, braucht der Administrator nur Paketweiterleitung auf den LVS-Routern aktivieren, damit die Anfragen ordnungsgemäß zwischen extern und den realen Servern geroutet werden. Werfen Sie einen Blick auf Abschnitt 2.5, „Aktivierung der Paketweiterleitung“ für Anweisungen zur Aktivierung von Paketweiterleitung. Weitere Konfiguration ist allerdings dann notwendig, wenn geclusterte Dienste mehr als einen Port benötigen, die demselben realen Server während einer Benutzersitzung zugeordnet werden sollen. Werfen Sie einen Blick auf Abschnitt 3.4, „Multiport-Dienste und das Lastverteilungs-Add-On“ für die Erstellung von Mehrportdiensten unter Verwendung von Firewall-Markierungen.

Sobald die Weiterleitung auf den LVS-Routern aktiviert und die realen Server eingerichtet sind und die geclusterten Dienste ausführen, verwenden Sie das **Piranha Configuration Tool**, um das Lastverteilungs-Add-On wie unter Kapitel 4, *Konfiguration des Lastverteilungs-Add-Ons mit dem **Piranha Configuration Tool*** aufgeführt zu konfigurieren.

Warnung

Konfigurieren Sie die Floating-IP für **eth0:1** oder **eth1:1** nicht, indem Sie Netzwerkskripte manuell bearbeiten oder ein Tool zur Konfiguration des Netzwerks verwenden. Verwenden Sie stattdessen das **Piranha Configuration Tool** wie unter Abschnitt 4.4, „**GLOBAL SETTINGS**“ und Abschnitt 4.6.1, „Der Unterabschnitt **VIRTUAL SERVER**“ aufgeführt.

Starten Sie nach Abschluss den **pulse**-Dienst, wie unter Abschnitt 4.8, „Das Lastverteilungs-Add-On starten“ dargestellt. Sobald **pulse** gestartet wurde und läuft, beginnt der aktive LVS-Router damit, Anfragen an den Pool der realen Server zu routen.

3.2. Das Lastverteilungs-Add-On via direktem Routing

Wie in Abschnitt 1.4.2, „Direktes Routing“ erwähnt, ermöglicht direktes Routing realen Servern, Pakete direkt an einen anfragenden Benutzer zu verarbeiten und zu routen, anstatt ausgehende Pakete durch den LVS-Router durchzuleiten. Direktes Routing setzt voraus, dass die realen Server physikalisch mit einem Netzwerksegment mit dem LVS-Router verbunden und in der Lage sind, auch ausgehende Pakete zu verarbeiten und weiterzuleiten.

Netzwerk-Layout

In einer Einstellung mit direktem Routing des Lastverteilungs-Add-Ons, muss der LVS-Router eingehende Anfragen erhalten können und sie an den entsprechenden realen Server zur Weiterverarbeitung routen. Die realen Server müssen dann die Antwort *direkt* an den Client routen. Wenn sich ein Client somit beispielsweise im Internet befindet, und die Pakete via LVS-Router an den realen Server schickt, muss der reale Server in der Lage sein, den Client direkt via Internet ansprechen zu können. Dies erreicht man, indem man ein Gateway für den realen Server einrichtet, das die Pakete an das Internet weiter gibt. Jeder reale Server im Server-Pool kann sein eigenes separates Gateway (und jedes Gateway seine eigene Internetverbindung) besitzen, was einen maximalen Durchsatz und eine maximale Skalierbarkeit ermöglicht. In typischen Lastverteilungs-Add-On-Einrichtungen können die realen Server jedoch via einem Gateway kommunizieren (und somit auch nur einer Netzwerkverbindung).

Wichtig

Es wird nicht empfohlen, den LVS-Router als ein Gateway für die realen Server zu verwenden, da dies zum einen die Komplexität der Einrichtung erhöht und zum anderen zu mehr Netzwerkauslastung auf dem LVS-Router führt, was wiederum den Engpass im Netzwerk darstellt, der auch beim NAT-Routing existiert.

Hardware

Die Hardware-Anforderungen eines Lastverteilungs-Add-On-Systems, das direktes Routing verwendet, ähneln den Anforderungen anderer Lastverteilungs-Add-Ons-Topologien. Während auf dem LVS-Router Red Hat Enterprise Linux laufen muss, um die eingehenden Anfragen zu verarbeiten und Lastverteilung für die realen Server durchzuführen, müssen die realen Server keine Linux-Maschinen sein, um ordnungsgemäß zu funktionieren. Die LVS-Router brauchen jeweils eine oder zwei NICs (abhängig davon, ob es einen Backup-Router gibt). Zur Erleichterung der Konfiguration und um den Datenverkehr deutlich zu unterscheiden, können Sie zwei NICs verwenden — eingehende Anfragen werden von einer NIC gehandhabt, sowie geroutete Pakete zu realen Servern auf der anderen.

Da die realen Server den LVS-Router übergehen und ausgehende Pakete direkt an einen Client versenden, wird ein Gateway zum Internet benötigt. Für die maximale Leistung und Verfügbarkeit kann jeder reale Server mit seinem eigenen Gateway verbunden sein, welches wiederum seine eigene, dedizierte Verbindung mit dem Trägernetzwerk besitzt, mit dem der Client verbunden ist (wie das Internet oder das Intranet).

Software

Es fällt einige Konfiguration über das **Piranha Configuration Tool** hinaus an, speziell für Administratoren, die mit ARP-Problemen konfrontiert werden, wenn sie das Lastverteilungs-Add-On via direktem Routing verwenden. Werfen Sie einen Blick auf Abschnitt 3.2.1, „Direktes Routing und **arptables_jf**“ oder Abschnitt 3.2.2, „Direktes Routing und **iptables**“ für mehr Informationen.

3.2.1. Direktes Routing und arptables_jf

Um direktes Routing mit Hilfe von **arptables_jf** zu konfigurieren, muss für jeden realen Server die virtuelle IP-Adresse konfiguriert sein, so dass diese Pakete direkt routen können. ARP-Anfragen für die VIP werden von den realen Servern komplett ignoriert und jegliche ARP-Pakete, die ansonsten gesendet werden könnten und die VIPs beinhalten, werden so manipuliert, dass sie statt der VIPs die IP des realen Servers enthalten.

Mit Hilfe der **arptables_jf**-Methode können sich Applikationen mit jeder einzelnen VIP oder jedem einzelnen Port binden, die der reale Server bedient. So gestattet die **arptables_jf**-Methode beispielsweise, dass mehrere Instanzen des Apache HTTP Server laufen und dabei explizit mit verschiedenen VIPs auf dem System gebunden sind. Die Verwendung von **arptables_jf** besitzt auch wesentliche Vorteile hinsichtlich der Leistungsfähigkeit gegenüber der Option **iptables**.

Allerdings können VIPs bei Verwendung der **arptables_jf**-Methode mit Hilfe von standardmäßigen Red Hat Enterprise Linux Systemkonfigurationstools nicht so konfiguriert werden, dass sie zum Zeitpunkt des Bootens starten.

Um jeden realen Server so zu konfigurieren, dass ARP-Anfragen für jede der virtuellen IP-Adressen ignoriert werden, führen Sie die folgenden Schritte durch:

1. Erstellen Sie die ARP-Tabelleneinträge für jede virtuelle IP-Adresse auf jedem realen Server (die real_ip ist die IP, die der "director" zur Kommunikation mit dem realen Server verwendet, oft die IP, die **eth0** zugeordnet ist)):

   ```
   arptables -A IN -d <virtual_ip> -j DROP
   arptables -A OUT -s <virtual_ip> -j mangle --mangle-ip-s <real_ip>
   ```

   Dies veranlasst die realen Server, alle ARP-Anfragen für die virtuellen IP-Adressen zu ignorieren und alle ausgehenden ARP-Antworten, die ansonsten evtl. die virtuelle IP enthalten könnten, zu verändern, so dass sie stattdessen die IP des Servers enthalten. Der einzige Knoten, der auf ARP-Anfragen für irgendeine der VIPs antworten sollte, ist der aktuell aktive LVS-Knoten.
2. Nachdem dies auf jedem realen Server abgeschlossen ist, speichern Sie die ARP-Tabelleneinträge, indem Sie die folgenden Befehle auf jedem realen Server eingeben:

   service arptables_jf save

   chkconfig --level 2345 arptables_jf on

   Der Befehl **chkconfig** veranlasst das System, die arptables-Konfiguration zum Zeitpunkt des

Bootens neu zu laden — vor dem Start des Netzwerks.

3. Konfigurieren Sie die virtuelle IP-Adresse auf allen realen Servern mit Hilfe von **ifconfig** zur Erstellung eines IP-Alias. Zum Beispiel:

```
# ifconfig eth0:1 192.168.76.24 netmask 255.255.252.0 broadcast
192.168.79.255 up
```

Oder unter Verwendung des **iproute2**-Hilfsprogramms **ip**, zum Beispiel:

```
# ip addr add 192.168.76.24 dev eth0
```

Wie bereits zuvor erwähnt, kann die virtuelle IP-Adresse mit Hilfe der Red Hat Systemkonfigurationstools nicht so konfiguriert werden, dass sie zum Zeitpunkt des Bootens startet. Eine Möglichkeit, dieses Problem zu umgehen, ist das Platzieren dieser Befehle in **/etc/rc.d/rc.local**.

4. Konfiguration von Piranha für direktes Routing. Werfen Sie einen Blick auf Kapitel 4, *Konfiguration des Lastverteilungs-Add-Ons mit dem **Piranha Configuration Tool*** für weitere Informationen.

3.2.2. Direktes Routing und iptables

Sie können das ARP-Problem ggf. auch umgehen, indem Sie die direkte Routing-Methode durch das Erstellen von **iptables** Firewall-Regeln verwenden. Um direktes Routing unter Verwendung von **iptables** zu konfigurieren, müssen Sie Regeln hinzufügen, die einen transparenten Proxy erstellen, so dass ein realer Server Pakete, die an die VIP-Adresse geschickt werden, bedient, auch wenn die VIP-Adresse nicht auf dem System existiert.

Die **iptables**-Methode ist einfacher zu konfigurieren, als die **arptables_jf**-Methode. Diese Methode umgeht außerdem das LVS-ARP-Problem komplett, da die virtuelle(n) IP-Adresse(n) nur auf dem aktiven LVS-Director existieren.

Allerdings treten bei der Verwendung der **iptables**-Methode Leistungsprobleme im Vergleich zu **arptables_jf** auf, da Overhead bei der Weiterleitung / beim Masquerading von jedem Paket auftritt.

Auch können Sie Ports nicht noch einmal verwenden, wenn Sie die **iptables**-Methode benutzen. Es ist beispielsweise nicht möglich, zwei separate Apache HTTP Server-Dienste auf Port 80 laufen zu lassen, da sich beide mit ***INADDR_ANY*** anstelle der virtuellen IP-Adresse binden müssen.

Um direkt Routing unter Verwendung der **iptables**-Methode zu konfigurieren, führen Sie die folgenden Schritte durch:

1. Führen Sie auf jedem realen Server den folgenden Befehl für jede VIP, jeden Port und jede Kombination von Protokollen (TCP oder UDP), die für die realen Server bedient werden sollen, aus:

   iptables -t nat -A PREROUTING -p <tcp|udp> -d <vip> --dport <port> -j REDIRECT

   Dieser Befehl veranlasst die realen Server, Pakete zu verarbeiten, die für die VIP und den spezifizierten Port bestimmt sind.
2. Speichern Sie die Konfiguration auf jedem realen Server:

```
# service iptables save
# chkconfig --level 2345 iptables on
```

Die oben aufgeführten Befehle veranlassen das System, die **iptables**-Konfiguration beim

Booten neu zu laden — bevor das Netzwerk gestartet wird.

3.3. Zusammensetzen der Konfiguration

Nach der Festlegung, welche der oben aufgeführten Routing-Methoden verwendet werden soll, sollte die Hardware miteinander im Netzwerk verbunden werden.

Wichtig

Die Adapter-Geräte auf den LVS-Routern müssen so konfiguriert werden, dass sie auf dieselben Netzwerke zugreifen. Falls **eth0** zum Beispiel mit dem öffentlichen Netzwerk und **eth1** mit dem privaten Netzwerk verbunden ist, dann müssen dieselben Geräte auf dem Backup-LVS mit denselben Netzwerken verbunden sein.
Auch wird das in der ersten Schnittstelle aufgelistete Gateway, das zum Zeitpunkt des Bootens aktiviert wird, zur Routing-Tabelle hinzugefügt und nachfolgende Gateways, die in anderen Schnittstellen aufgeführt sind, werden ignoriert. Dies muss besonders bei der Konfiguration neuer Server bedacht werden.

Nachdem Sie die Hardware physikalisch miteinander verbunden haben, konfigurieren Sie die Netzwerkschnittstellen auf dem primären und den Backup-LVS-Routern. Hierzu können Sie eine grafische Applikation wie **system-config-network** verwenden, oder die Netzwerkskripte manuell bearbeiten. Werfen Sie einen Blick auf das Kapitel *Netzwerkkonfiguration* im *Red Hat Enterprise Linux Bereitstellungshandbuch* für weitere Informationen über das Hinzufügen von Geräten mit Hilfe von **system-config-network**. Was den Rest des Kapitels angeht, werden Änderungen an Netzwerkschnittstellen entweder manuell oder durch das **Piranha Configuration Tool** durchgeführt.

3.3.1. Allgemeine Tipps bezüglich Lastverteilungs-Add-On-Vernetzung

Konfigurieren Sie die realen IP-Adressen sowohl für die öffentlichen, als auch die privaten Netzwerke auf den LVS-Routern, bevor Sie versuchen, das Lastverteilungs-Add-On mit dem **Piranha Configuration Tool** zu konfigurieren. Die Abschnitte zu jeder Topologie liefern jeweils Beispiel-Netzwerkadressen, es werden jedoch die tatsächlichen Adressen benötigt. Nachfolgend sind ein paar nützliche Befehle aufgeführt, um Netzwerkschnittstellen zu aktivieren oder ihren Status zu überprüfen.

Aktivieren von realen Netzwerkschnittstellen

Um eine reale Netzwerkschnittstelle zu aktivieren, führen Sie den folgenden Befehl als Root aus und ersetzen dabei ***N*** mit der entsprechenden Nummer der Schnittstelle (**eth0** und **eth1**).

```
/sbin/ifup ethN
```

Warnung

Verwenden Sie die **ifup**-Skripte *nicht*, um jegliche floating IP-Adressen zu aktivieren und für deren Konfiguration Sie **Piranha Configuration Tool** verwenden können (**eth0:1** oder **eth1:1**). Verwenden Sie den Befehl **service**, um stattdessen **pulse** zu starten (werfen Sie einen Blick auf Abschnitt 4.8, „Das Lastverteilungs-Add-On starten“ für Details).

Deaktivieren von Schnittstellen des realen Netzwerks

Um eine reale Netzwerkschnittstelle zu deaktivieren, führen Sie den folgenden Befehl als Root

aus und ersetzen dabei ***N*** mit der entsprechenden Nummer der Schnittstelle (**eth0** und **eth1**).

```
/sbin/ifdown ethN
```

Überprüfen des Status von Netzwerkschnittstellen

Falls Sie zu jedem beliebigen Zeitpunkt überprüfen müssen, welche Netzwerkschnittstellen aktiviert sind, geben Sie Folgendes ein:

```
/sbin/ifconfig
```

Um die Routing-Tabelle für eine Maschine anzusehen, führen Sie den folgenden Befehl aus:

```
/sbin/route
```

3.4. Multiport-Dienste und das Lastverteilungs-Add-On

LVS-Router aus jeder Topologie erfordern bei der Erstellung von Multiport-Lastverteilungs-Add-On-Diensten zusätzliche Konfiguration. Multiport-Dienste können künstlich erstellt werden, indem Firewall-Markierungen verwendet werden, um unterschiedliche, jedoch ähnliche Protokolle zusammenzufassen, wie beispielsweise HTTP (Port 80) und HTTPS (Port 443) oder wenn das Lastverteilungs-Add-On mit echten Multiport-Protokollen, wie FTP, verwendet wird. In beiden Fällen verwendet der LVS-Router Firewall-Markierungen, um zu erkennen, dass Pakete, die für unterschiedliche Ports bestimmt sind, jedoch dieselben Firewall-Markierungen tragen, identisch gehandhabt werden sollten. Wenn sie außerdem mit Persistenz kombiniert werden, stellen Firewall-Markierungen sicher, dass Verbindungen von der Client-Maschine zum selben Host geroutet werden, so lange die Verbindungen im Rahmen der Zeitspanne anfallen, die vom Persistenz-Parameter spezifiziert wird. Werfen Sie einen Blick auf Abschnitt 4.6.1, „Der Unterabschnitt **VIRTUAL SERVER**“ für weitere Informationen hinsichtlich der Zuweisung von Persistenz an einen virtuellen Server.

Leider kann der Mechanismus, der für die Lastverteilung auf den realen Servern verwendet wird — IPVS — zwar die Firewall-Markierungen erkennen, die einem Paket zugeordnet sind, kann jedoch nicht selbst Firewall-Markierungen zuweisen. Der Job der *Zuweisung* von Firewall-Markierungen muss vom Netzwerk-Paketfilter **iptables** außerhalb des **Piranha Configuration Tool** durchgeführt werden.

3.4.1. Zuweisen von Firewall-Markierungen

Um einem Paket, das für einen bestimmten Port gedacht ist, Firewall-Markierungen zuzuweisen, muss der Administrator **iptables** verwenden.

Dieser Abschnitt erläutert, wie z.B. HTTP und HTTPS zusammengefasst werden können, allerdings ist auch FTP ein weiteres, häufig geclustertes Multiport-Protokoll. Falls ein Lastverteilungs-Add-On für FTP-Dienste verwendet wird, werfen Sie einen Blick auf Abschnitt 3.5, „Konfiguration von FTP“ für Konfigurationsdetails.

Als Grundregel bei der Verwendung von Firewall-Markierungen kann festgehalten werden, dass für jedes Protokoll, das eine Firewall-Markierung in **Piranha Configuration Tool** verwendet, eine entsprechende **iptables**-Regel existieren muss, um den Netzwerkpaketen Markierungen zuzuweisen.

Stellen Sie vor der Erstellung von Netzwerk-Paketfilterregeln sicher, dass nicht bereits Regeln existieren. Öffnen Sie hierfür einen Shell-Prompt, loggen Sie sich als Root ein und geben Sie ein:

```
/sbin/service iptables status
```

Falls **iptables** nicht läuft, erscheint der Prompt umgehend wieder.

Falls **iptables** aktiv ist, werden eine Reihe an Regeln angezeigt.

/sbin/service iptables stop

Falls die bereits existierenden Regeln wichtig sind, überprüfen Sie den Inhalt von **/etc/sysconfig/iptables** und kopieren Sie sämtliche Regeln, die es wert sind, zu speichern, an einen sicheren Ort, bevor Sie fortfahren.

Nachfolgend sind Regeln aufgeführt, die dieselbe Firewall-Markierung, 80, eingehendem Datenverkehr, der für die floating IP,***n.n.n.n***, auf den Ports 80 und 443, zuweisen

/sbin/modprobe ip_tables

/sbin/iptables -t mangle -A PREROUTING -p tcp -d *n.n.n.n*/32 --dport 80 -j MARK --set-mark 80

/sbin/iptables -t mangle-A PREROUTING -p tcp -d *n.n.n.n*/32 --dport 443 -j MARK --set-mark 80

Für Anweisungen zur Zuweisung der VIP für die öffentlich Netzwerkschnittstelle, werfen Sie einen Blick auf Abschnitt 4.6.1, „Der Unterabschnitt **VIRTUAL SERVER**“. Beachten Sie bitte weiterhin, dass Sie als Root eingeloggt sein müssen und das Modul für **iptables** geladen sein muss, bevor Regeln das erste Mal erstellt werden.

In den oben aufgeführten **iptables**-Befehlen sollte ***n.n.n.n*** durch die floating IP für Ihre virtuellen HTTP- und HTTPS-Server ersetzt werden. Diese Befehle haben den reinen Effekt, dass jeglichem Datenverkehr, der an die VIP auf dem entsprechenden Port gerichtet ist, eine Firewall-Markierung von 80 zugewiesen wird, was im Gegenzug von IPVS erkannt und entsprechend weitergeleitet wird.

Warnung

Die oben aufgeführten Befehle werden unmittelbar umgesetzt, bleiben jedoch bei einem Neustart des Systems nicht bestehen. Um sicherzustellen, dass Einstellungen zum Netzwerkpaketfilter bei einem Neustart wiederhergestellt werden, werfen Sie einen Blick auf Abschnitt 3.6, „Speichern der Einstellungen des Netzwerkpaketfilters“.

3.5. Konfiguration von FTP

Das File Transport Protocol (FTP) ist ein altes und komplexes Multiport-Protokoll, welches eine Reihe von Herausforderungen in einer Lastverteilungs-Add-On-Umgebung mit sich bringt. Um die Art dieser Herausforderungen zu verstehen, müssen Sie zunächst ein paar zentrale Dinge über die Funktionsweise von FTP verstehen.

3.5.1. Funktionsweise von FTP

Bei den meisten anderen Server-Client-Beziehungen öffnet die Client-Maschine eine Verbindung mit dem Server auf einem bestimmten Port und der Server antwortet dem Client anschließend auf diesem Port. Wenn sich ein FTP-Client mit einem FTP-Server verbindet, wird eine Verbindung auf dem FTP-Kontrollport 21 geöffnet. Der *Client* teilt dem FTP-*Server* dann mit, ob eine *aktive* oder *passive* Verbindung hergestellt werden soll. Die vom Client gewählte Verbindung bestimmt die Antwort des Servers und auf welchen Ports Transaktionen stattfinden.

Die beiden Arten von Datenverbindungen lauten:

Aktive Verbindungen

Beim Aufbau einer aktiven Verbindung öffnet der *Server* eine Datenverbindung mit dem Client von Port 20 bis zu einem Port in einem hohen Bereich auf der Client-Maschine. Alle Daten vom Server werden dann über diese Verbindung weitergeleitet.

Passive Verbindungen

Beim Aufbau einer passiven Verbindung fragt der *Client* beim FTP-Server bezüglich des Aufbaus eines Ports für eine passive Verbindung an, der sich auf jedem Port größer als 10,000 befinden kann. Der Server wird dann an diesen großzahligen Port für diese spezielle Sitzung gebunden und leitet die Portnummer zurück an den Client. Der Client öffnet dann den neu gebundenen Port für die Datenverbindung. Jede Datenanfrage, die der Client durchführt, resultiert in einer separaten Datenverbindung. Die meisten modernen FTP-Clients versuchen bei der Abfrage von Daten von Servern eine passive Verbindung aufzubauen.

Anmerkung

Der *Client* ermittelt den Verbindungstyp, nicht der Server. Dies bedeutet, dass der LVS-Router so konfiguriert werden muss, dass er sowohl aktive, als auch passive Verbindungen unterstützt, um FTP effektiv zu clustern.
Die Verbindung FTP-Client/Server kann potentiell eine große Anzahl an Ports öffnen, die dem **Piranha Configuration Tool** und IPVS nicht bekannt sind.

3.5.2. Auswirkungen auf das Lastverteilungs-Add-On-Routing

IPVS-Paketweiterleitung gestattet lediglich Verbindungen in das Cluster und aus dem Cluster heraus, basierend auf der Erkennung ihrer Portnummer oder Firewall-Markierungen. Falls ein Client außerhalb des Clusters versucht, einen Port zu öffnen, den die aktuelle Konfiguration von IPVS nicht umfasst, wird die Verbindung abgebrochen. Falls der reale Server gleichermaßen versucht, eine Verbindung auf einem Port, den IPVS nicht kennt zurück zum Internet herzustellen, wird die Verbindung abgebrochen. Dies bedeutet, dass *alle* Verbindungen von FTP-Clients aus dem Internet dieselbe Firewall-Markierung zugewiesen haben *müssen* und alle Verbindungen vom FTP-Server ordnungsgemäß zum Internet unter Verwendung von Netzwerk-Paketfilterregeln weitergeleitet werden *müssen*.

3.5.3. Erstellen von Netzwerk-Paketfilterregeln

Bevor Sie irgendwelche **`iptables`**-Regeln für den FTP-Dienst zuweisen, sehen Sie noch einmal die Informationen bezüglich Multiport-Diensten und Techniken zur Überprüfung von existierenden Netzwerk-Paketfilterregeln unter Abschnitt 3.4.1, „Zuweisen von Firewall-Markierungen“ durch.

Nachfolgend sind Regeln aufgeführt, die dieselbe Firewall-Markierung, 21, FTP-Datenverkehr zuweisen. Damit diese Regeln ordnungsgemäß funktionieren, müssen Sie außerdem den Unterabschnitt **`VIRTUAL SERVER`** des **Piranha Configuration Tool** verwenden, um einen virtuellen Server für Port 21 mit einem Wert von **`21`** im Feld **`Firewall Mark`** zu konfigurieren. Werfen Sie einen Blick auf Abschnitt 4.6.1, „Der Unterabschnitt **`VIRTUAL SERVER`**“ für Details.

3.5.3.1. Regeln für aktive Verbindungen

Die Regeln für aktive Verbindungen teilen dem Kernel mit, Verbindungen zu akzeptieren und weiterzuleiten, die auf der *internen* floating IP-Adresse auf Port 20 ankommen — dem FTP-Datenport.

Der folgende **iptables**-Befehl ermöglicht dem LVS-Router, ausgehende Verbindungen von realen Servern, über die das IPVS nichts weiß, zu akzeptieren:

```
/sbin/iptables -t nat -A POSTROUTING -p tcp -s n.n.n.0/24 --sport 20 -j MASQUERADE
```

Im **iptables**-Befehl sollte ***n.n.n*** mit den ersten drei Werten der floating IP für die interne Netzwerkschnittstelle der NAT-Schnittstelle ersetzt werden, wie im **GLOBAL SETTINGS**-Panel des **Piranha Configuration Tool** definiert.

3.5.3.2. Regeln für passive Verbindungen

Die Regeln für passive Verbindungen weisen die entsprechende Firewall-Markierung für Verbindungen zu, die aus dem Internet an die floating IP für den Dienst auf einer großen Bandbreite an Ports eingehen — 10,000 to 20,000.

Warnung

Falls Sie die Port-Bandbreite für passive Verbindungen einschränken, müssen Sie auch den VSFTP-Server so konfigurieren, dass er eine passende Port-Bandbreite verwendet. Dies erreicht man, indem folgende Zeilen zu **/etc/vsftpd.conf** hinzugefügt werden:

```
pasv_min_port=10000
pasv_max_port=20000
```

Sie müssen auch die Adresse kontrollieren, die der Server dem Client für passive FTP-Verbindungen anzeigt. Fügen Sie in einem NAT-gerouteten Lastverteilungs-Add-On-System die folgende Zeile zu **/etc/vsftpd.conf** hinzu, um die IP-Adresse des realen Servers mit der des VIP außer Kraft zu setzen, welche diejenige ist, die ein Client bei einer Verbindung sieht. Zum Beispiel:

```
pasv_address=n.n.n.n
```

Ersetzen Sie ***n.n.n.n*** mit der VIP-Adresse des LVS-Systems.

Konsultieren Sie zur Konfiguration weiterer FTP-Server die entsprechende Dokumentation.

Diese Bandbreite sollte für die meisten Situationen ausreichen. Sie können die Zahl jedoch auch so erhöhen, dass alle verfügbaren nicht-sicheren Ports enthalten sind, indem Sie **10000:20000** im unten aufgeführten Befehl in **1024:65535** ändern.

Die folgenden **iptables**-Befehle haben den reinen Effekt, dass jeglichem Datenverkehr, der an die floating IP auf dem entsprechenden Port gerichtet ist, eine Firewall-Markierung von 21 zugewiesen wird, was im Gegenzug von IPVS erkannt und entsprechend weitergeleitet wird:

```
/sbin/iptables -t mangle -A PREROUTING -p tcp -d n.n.n.n/32 --dport 21 -j MARK --set-mark 21
```

```
/sbin/iptables -t mangle -A PREROUTING -p tcp -d n.n.n.n/32 --dport 10000:20000 -j MARK --set-mark 21
```

In den **iptables**-Befehlen sollte ***n.n.n.n*** durch die floating IP für den virtuellen FTP-Server ersetzt werden, wie im Unterabschnitt **VIRTUAL SERVER** des **Piranha Configuration Tool** definiert.

Warnung

Die oben aufgeführten Befehle werden unmittelbar umgesetzt, bleiben jedoch bei einem Neustart des Systems nicht bestehen. Um sicherzustellen, dass Einstellungen zum Netzwerkpaketfilter bei einem Neustart wiederhergestellt werden, werfen Sie einen Blick auf Abschnitt 3.6, „Speichern der Einstellungen des Netzwerkpaketfilters“

Abschließend müssen Sie sicherstellen, dass der entsprechende Dienst im jeweiligen Runlevel auf aktiv gesetzt ist. Werfen Sie einen Blick auf Abschnitt 2.1, „Konfiguration von Diensten auf den LVS-Routern“ für weitere Informationen diesbezüglich.

3.6. Speichern der Einstellungen des Netzwerkpaketfilters

Speichern Sie die Einstellungen nach der Konfiguration der entsprechenden Netzwerkpaketfilter für Ihre Situation, damit diese bei einem Neustart wieder hergestellt werden. Geben Sie den folgenden Befehl für **iptables** ein:

```
/sbin/service iptables save
```

Dies speichert die Einstellungen in **/etc/sysconfig/iptables**, so dass sie zum Zeitpunkt des Bootens wieder abgerufen werden können.

Sobald diese Datei gespeichert wurde, können Sie den Befehl **/sbin/service** verwenden, um **iptables** zu starten, zu stoppen und den Status zu überprüfen. **/sbin/service** lädt das entsprechende Modul automatisch für Sie. Werfen Sie einen Blick auf Abschnitt 2.3, „Starten des **Piranha Configuration Tool** Dienstes“ für ein Beispiel zur Anwendung des Befehls **/sbin/service**.

Abschließend müssen Sie sicherstellen, dass der entsprechende Dienst in den jeweiligen Runleveln auf aktiv gesetzt ist. Werfen Sie einen Blick auf Abschnitt 2.1, „Konfiguration von Diensten auf den LVS-Routern“ für weitere Informationen diesbezüglich.

Im nächsten Kapitel wird erklärt, wie das **Piranha Configuration Tool** verwendet wird, um den LVS-Router zu konfigurieren und beschreibt die notwendigen Schritte, um Lastverteilungs-Add-On zu aktivieren.

Kapitel 4. Konfiguration des Lastverteilungs-Add-Ons mit dem Piranha Configuration Tool

Das **Piranha Configuration Tool** liefert einen strukturierten Ansatz zur Erstellung der nötigen Konfigurationsdatei für das Lastverteilungs-Add-On — **`/etc/sysconfig/ha/lvs.cf`**. Dieses Kapitel beschreibt die grundlegenden Funktionen des **Piranha Configuration Tool** und der Aktivierung des Lastverteilungs-Add-Ons nach Abschluss der Konfiguration.

Wichtig

Die Konfigurationsdatei für das Lastverteilungs-Add-On befolgt strenge Formatierungsregeln. Das Verwenden des **Piranha Configuration Tool** ist der beste Weg, syntaktische Fehler in **`lvs.cf`** und damit Software-Ausfälle zu vermeiden.

4.1. Benötigte Software

Der Dienst **`piranha-gui`** muss auf dem primären LVS-Router laufen, um das **Piranha Configuration Tool** benutzen zu können. Um das Lastverteilungs-Add-On konfigurieren zu können, brauchen Sie mindestens einen Text-Browser, wie **`links`**. Falls Sie von einer anderen Maschine aus auf den LVS-Router zugreifen, brauchen Sie außerdem eine **ssh**-Verbindung mit dem primären LVS-Router als Root-Benutzer.

Während der Konfiguration des primären LVS-Routers ist es eine gute Idee, parallel eine **ssh**-Verbindung in einem Terminal-Fenster offen zu haben. Diese Verbindung liefert eine sichere Möglichkeit, **`pulse`** und andere Dienste neu zu starten, Netzwerk-Paketfilter zu konfigurieren und **`/var/log/messages`** auf der Suche nach Fehlern zu überwachen.

Die nächsten vier Abschnitte führen Sie durch jede der Konfigurationsseiten des **Piranha Configuration Tool** und geben Anweisungen, wie es zur Einrichtung des Lastverteilungs-Add-Ons verwendet werden kann.

4.2. Einloggen in das Piranha Configuration Tool

Bei der Konfiguration des Lastverteilungs-Add-Ons sollten Sie immer mit der Konfiguration des primären Routers mit dem **Piranha Configuration Tool** beginnen. Stellen Sie hierfür sicher, dass der **`piranha-gui`**-Dienst ausgeführt wird und ein administratives Passwort gesetzt wurde, wie unter Abschnitt 2.2, „Einrichten eines Passworts für das **Piranha Configuration Tool**“ beschrieben.

Wenn Sie von lokal aus auf die Maschine zugreifen, können Sie **`http://localhost:3636`** in einem Web-Browser öffnen, um auf das **Piranha Configuration Tool** zuzugreifen. Geben Sie ansonsten den Hostnamen oder die reale IP-Adresse für den Server ein, gefolgt von **`:3636`**. Sobald sich der Browser verbindet, sehen Sie einen Bildschirm, wie unter Abbildung 4.1, „Das Willkommens-Panel“ angezeigt.

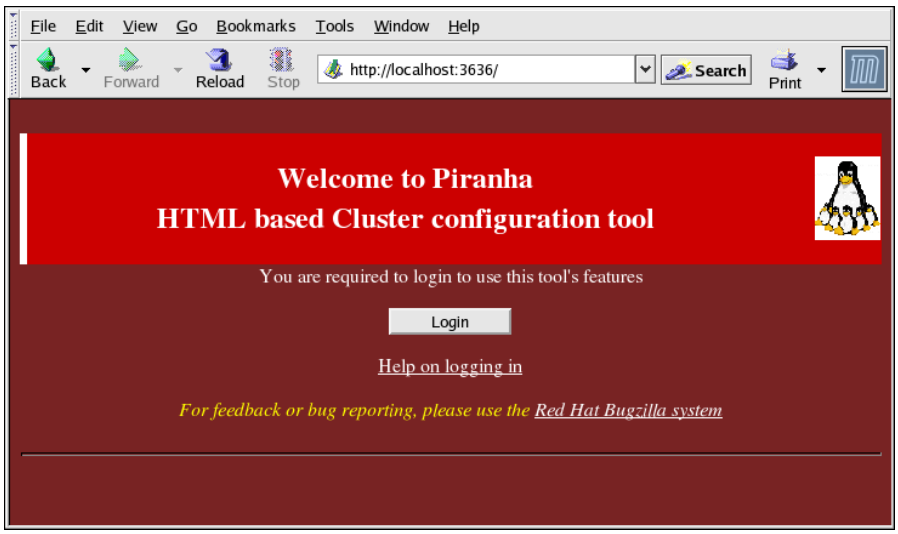
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://localhost:3636/
Search
Print
Welcome to Piranha
HTML based Cluster configuration tool
You are required to login to use this tool's features
Login
Help on logging in
For feedback or bug reporting, please use the Red Hat Bugzilla system

Abbildung 4.1. Das Willkommens-Panel

Klicken Sie auf die Schaltfläche **Login** und geben Sie **piranha** als **Username** und das administrative Passwort, dass Sie im Feld **Password** erstellt haben, ein.

Das **Piranha Configuration Tool** besteht aus vier Haupt-Bildschirmen oder *Panels*. Zusätzlich enthält das Panel **VIRTUAL SERVERS** vier *Unterabschnitte*. Das Panel **CONTROL/MONITORING** ist das erste Panel nach der Login-Seite.

4.3. CONTROL/MONITORING

Das Panel **CONTROL/MONITORING** liefert einen eingeschränkten Laufzeit-Status des Lastverteilungs-Add-Ons. Es zeigt den Status des **pulse**-Daemon, der LVS-Routing-Tabelle und den von LVS erzeugten **nanny**-Prozessen an.

Anmerkung

Die Felder für **CURRENT LVS ROUTING TABLE** und **CURRENT LVS PROCESSES** bleiben leer, bis Sie das Lastverteilungs-Add-On tatsächlich starten, wie unter Abschnitt 4.8, „Das Lastverteilungs-Add-On starten“ gezeigt.

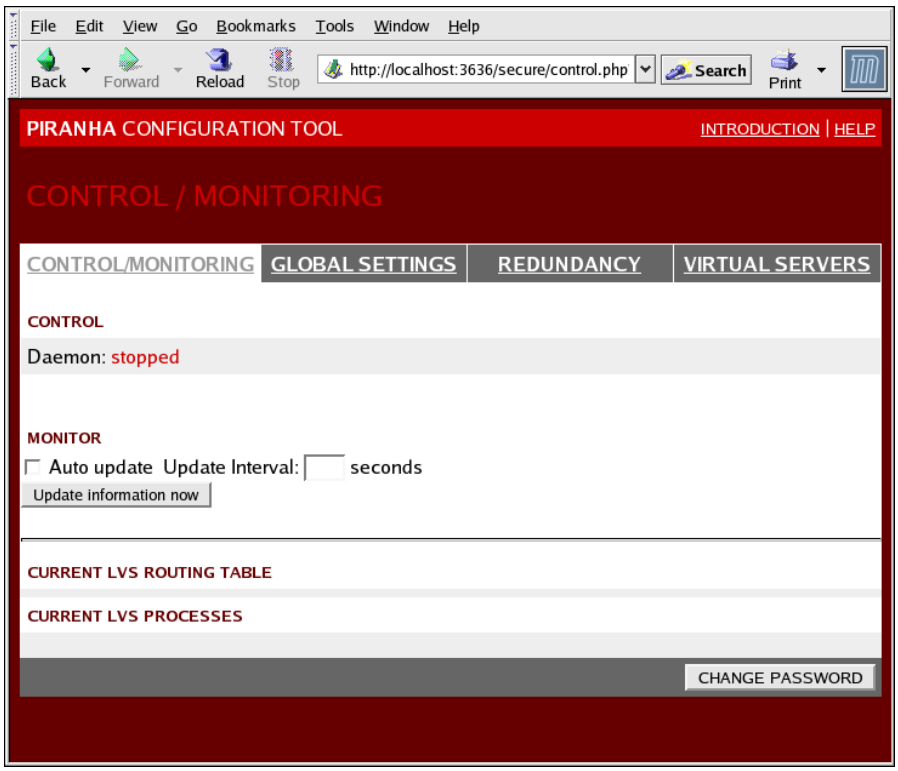

Abbildung 4.2. Das Panel `CONTROL/MONITORING`

`Auto update`

Die Statusanzeige dieser Seite kann zu einem vom Benutzer konfigurierten Intervall automatisch aktualisiert werden. Klicken Sie auf **`Auto update`** und setzen Sie die gewünschte Aktualisierungsfrequenz im Textfeld **`Update frequency in seconds`** (der Standardwert ist 10 Sekunden).

Es wird nicht empfohlen, die automatische Aktualisierung auf ein Zeitintervall von weniger als 10 Sekunden zu setzen. Andernfalls kann es problematisch werden, das Zeitintervall für die **`Auto update`** neu zu konfigurieren, da die Seite zu schnell neu lädt. Falls Sie dieses Problem haben, klicken Sie einfach auf ein anderes Panel und dann zurück auf **`CONTROL/MONITORING`**.

Das **`Auto update`**-Feature funktioniert nicht mit allen Browsern, wie beispielsweise **Mozilla**.

`Update information now`

Sie können die Statusinformationen manuell aktualisieren, indem Sie auf diese Schaltfläche klicken.

`CHANGE PASSWORD`

Ein Klick auf diese Schaltfläche führt Sie zu einer Hilfeseite mit Informationen, wie Sie das administrative Passwort für das **Piranha Configuration Tool** ändern können.

4.4. `GLOBAL SETTINGS`

Mit Hilfe des Panels **`GLOBAL SETTINGS`** definieren Sie die Netzwerkdetails für die öffentlichen und

privaten Netzwerkschnittstellen des primären LVS-Routers.

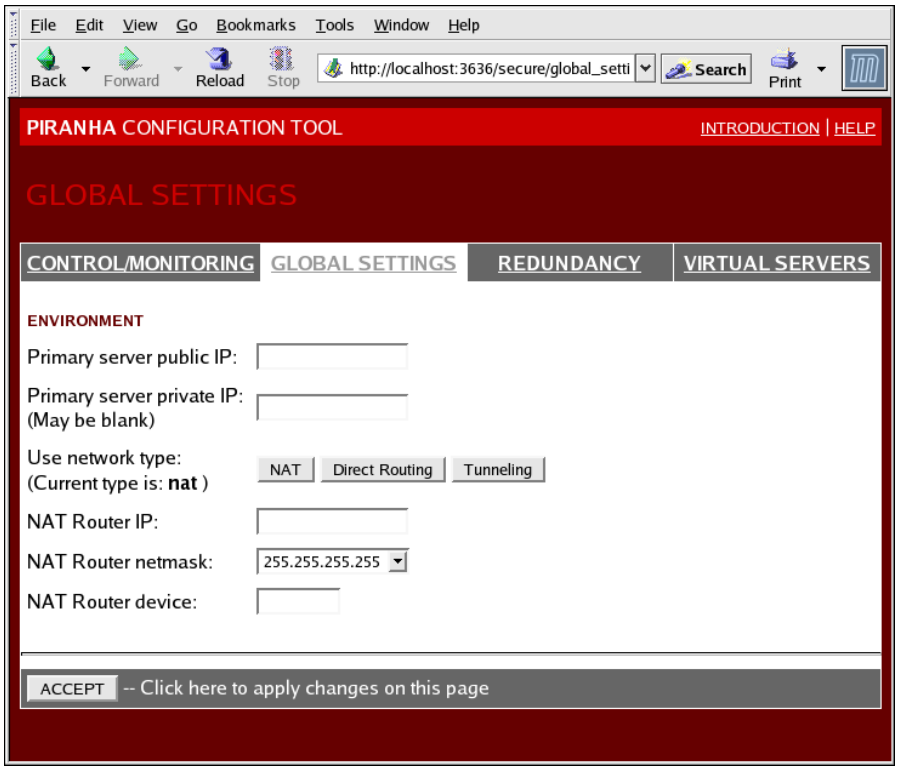

Abbildung 4.3. Das Panel `GLOBAL SETTINGS`

In der oberen Hälfte dieses Panels werden die öffentlichen und privaten Netzwerkschnittstellen des primären LVS-Routers eingerichtet. Dies sind die bereits unter Abschnitt 3.1.1, „Konfiguration von Netzwerkschnittstellen für das Lastverteilungs-Add-On mit NAT“ konfigurierten Schnittstellen.

`Primary server public IP`

Geben Sie in diesem Feld die öffentlich routbare reale IP-Adresse für den primären LVS-Knoten ein.

`Primary server private IP`

Geben Sie die reale IP-Adresse für eine alternative Netzwerkschnittstelle auf dem primären LVS-Knoten ein. Diese Adresse wird ausschließlich als alternativer Heartbeat-Kanal für den Backup-Router verwendet und muss nicht mit der unter Abschnitt 3.1.1, „Konfiguration von Netzwerkschnittstellen für das Lastverteilungs-Add-On mit NAT“ zugewiesenen realen privaten IP-Adresse zusammenhängen. Sie können dieses Feld auch leer lassen, was allerdings zur Folge hat, dass es keinen alternativen Heartbeat-Kanal gibt, den der Backup-LVS-Router verwenden kann, was somit in einem "Single Point of Failure" resultiert.

Anmerkung

Die private IP wird nicht für Konfigurationen, die **`Direct Routing`** verwenden, benötigt, da alle realen Server, wie auch die LVS-Directors dieselbe virtuelle IP-Adresse teilen und dieselbe IP-Route-Konfiguration besitzen sollten.

Anmerkung

Die private IP des primären LVS-Router kann auf jeder Schnittstelle konfiguriert werden, die TCP/IP akzeptiert, unabhängig davon, ob es ein Ethernet-Adapter oder ein serieller Port ist.

`Use network type`

Klicken Sie auf die Schaltfläche **`NAT`**, um NAT-Routing auszuwählen.

Klicken Sie auf die Schaltfläche **`Direct Routing`**, um direktes Routing auszuwählen.

Die nächsten drei Felder behandeln speziell die virtuellen Netzwerkschnittstellen des NAT-Routers, die das private Netzwerk mit den realen Servern verbinden. Diese Felder sind *nicht* für den Netzwerktyp "Direktes Routing" relevant.

`NAT Router IP`

Geben Sie die private Floating-IP in dieses Textfeld ein. Diese Floating-IP sollte als Gateway für die realen Server verwendet werden.

`NAT Router netmask`

Falls die Floating-IP des NAT-Routers eine spezielle Netzmaske benötigt, wählen Sie diese aus der Drop-Down-Liste.

`NAT Router device`

Verwenden Sie dieses Textfeld, um den Gerätenamen der Netzwerkschnittstelle für die floating IP-Adresse zu definieren, wie beispielsweise **`eth1:1`**.

Anmerkung

Sie sollten einen Alias für die NAT floating IP-Adresse mit der Ethernet-Schnittstelle, die mit dem privaten Netzwerk verbunden ist, kreieren. In diesem Beispiel befindet sich das private Netzwerk auf der **`eth1`**-Schnittstelle, so dass **`eth1:1`** die floating IP-Adresse darstellt.

Warnung

Klicken Sie nach Fertigstellung dieser Seite auf die Schaltfläche **`ACCEPT`** um sicherzustellen, dass Sie keine Änderungen verlieren, wenn Sie ein neues Panel auswählen.

4.5. REDUNDANCY

Das Panel **`REDUNDANCY`** ermöglicht Ihnen die Konfiguration des Backup-LVS-Router-Knotens und das

Einstellen verschiedener Heartbeat-Überwachungsoptionen.

Anmerkung

Wenn Sie diesen Bildschirm das erste Mal aufrufen, wird ein Status "inaktiv" für **Backup** und eine Schaltfläche **ENABLE** angezeigt. Klicken Sie auf die Schaltfläche **ENABLE**, um den Backup-LVS-Router zu konfigurieren, so dass der Bildschirm der Abbildung 4.4, „Das Panel **REDUNDANCY**“ entspricht.

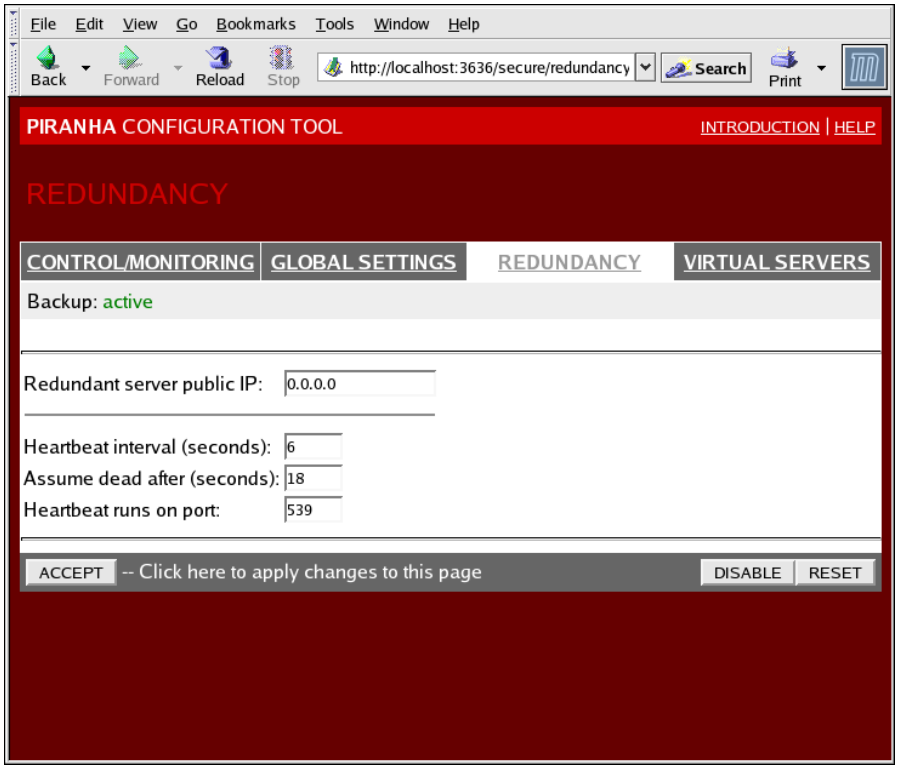

Abbildung 4.4. Das Panel REDUNDANCY

Redundant server public IP

Geben Sie die öffentliche reale IP-Adresse für den Backup-LVS-Router-Knoten an.

Redundant server private IP

Geben Sie in diesem Textfeld die private reale IP-Adresse für den Backup-Knoten an.

Falls Sie das Feld **Redundant server private IP** nicht sehen können, gehen Sie zurück zum Panel **GLOBAL SETTINGS**, geben Sie eine **Primary server private IP** Adresse ein und klicken Sie auf **ACCEPT**.

Der Rest des Panels ist für die Konfiguration des Heartbeat-Kanals gedacht, welcher von dem Backup-Knoten zur Überwachung des primären Knotens hinsichtlich eines Ausfall verwendet wird.

Heartbeat Interval (seconds)

In diesem Feld werden die Anzahl der Sekunden zwischen den Heartbeats gesetzt — Dies ist das Intervall, in dem der Backup-Knoten den funktionalen Status des primären LVS-Knoten

überprüft.

`Assume dead after (seconds)`

Falls der primäre LVS-Knoten nicht nach dieser Anzahl an Sekunden antwortet, leitet der Backup-LVS-Knoten die Ausfallsicherung ein.

`Heartbeat runs on port`

In diesem Feld wird der Port eingestellt, auf dem der Heartbeat mit dem primären LVS-Knoten kommuniziert. Der Standardwert beträgt 539, falls dieses Feld leer gelassen wird.

Warnung

Denken Sie nach dem Ändern dieser Seite daran, auf die Schaltfläche **`ACCEPT`** zu klicken, um sicherzustellen, dass Sie keine Änderungen verlieren, wenn Sie ein neues Panel auswählen.

4.6. `VIRTUAL SERVERS`

Das Panel **`VIRTUAL SERVERS`** zeigt Informationen zu jedem derzeit definierten virtuellen Server an. Jeder Tabelleneintrag zeigt den Status des virtuellen Servers, dessen Namen, die dem Server zugeteilte virtuelle IP, die Netzmaske der virtuellen IP, die Portnummer, auf der der Dienst kommuniziert, das verwendete Protokoll und die Schnittstelle des virtuellen Geräts.

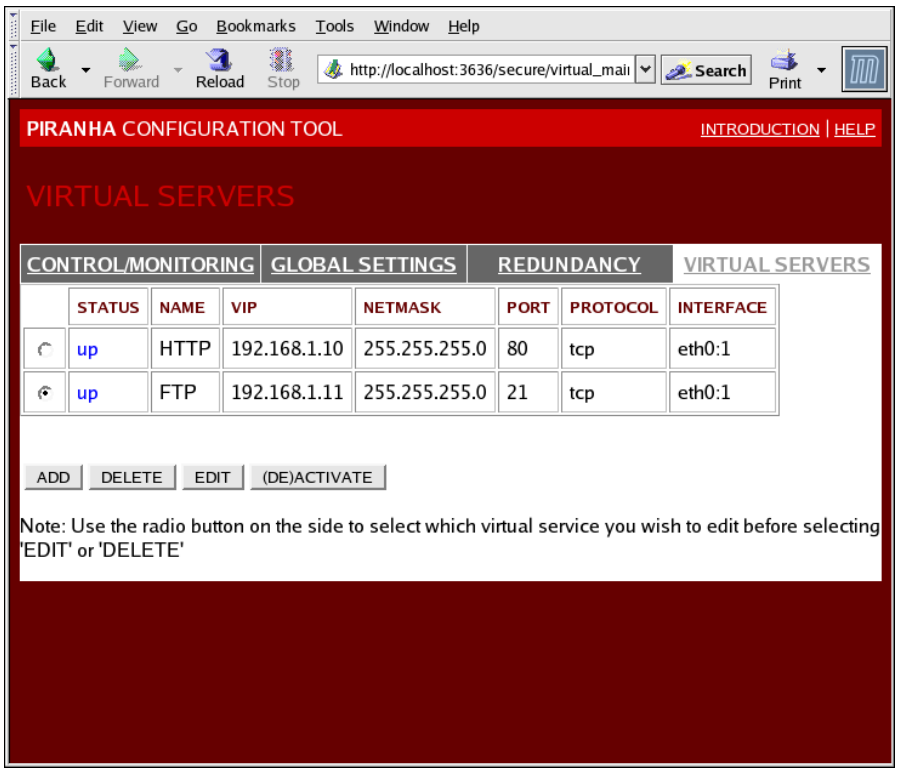
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://localhost:3636/secure/virtual_mai
Search
Print
PIRANHA CONFIGURATION TOOL
INTRODUCTION | HELP
VIRTUAL SERVERS
CONTROL/MONITORING
GLOBAL SETTINGS
REDUNDANCY
VIRTUAL SERVERS
STATUS NAME VIP NETMASK PORT PROTOCOL INTERFACE
up HTTP 192.168.1.10 255.255.255.0 80 tcp eth0:1
up FTP 192.168.1.11 255.255.255.0 21 tcp eth0:1
ADD DELETE EDIT (DE)ACTIVATE
Note: Use the radio button on the side to select which virtual service you wish to edit before selecting 'EDIT' or 'DELETE'

Abbildung 4.5. Das Panel `VIRTUAL SERVERS`

Jeder Server, der im Panel **`VIRTUAL SERVERS`** angezeigt wird, kann auf den nachfolgenden

Bildschirmen oder *Unterabschnitten* konfiguriert werden.

Um einen Dienst hinzuzufügen, klicken Sie auf die Schaltfläche **ADD**. Um einen Dienst zu entfernen, wählen Sie diesen aus, indem Sie auf das Auswahlsymbol neben dem virtuellen Server und dann auf die Schaltfläche **DELETE** klicken.

Um einen virtuellen Server in der Tabelle zu aktivieren oder zu deaktivieren, klicken Sie auf dessen Auswahlsymbol und anschließend auf die Schaltfläche **(DE)ACTIVATE**.

Nach Hinzufügen eines virtuellen Servers, können Sie diesen konfigurieren, indem Sie auf das Auswahlsymbol links daneben und anschließend auf die Schaltfläche **EDIT** klicken, um den Unterabschnitt **VIRTUAL SERVER** anzuzeigen.

4.6.1. Der Unterabschnitt VIRTUAL SERVER

Das Panel **VIRTUAL SERVER** im Unterabschnitt, wie unter Abbildung 4.6, „Der Unterabschnitt **VIRTUAL SERVERS**“ gezeigt, ermöglicht Ihnen die Konfiguration eines einzelnen virtuellen Servers. Verknüpfungen zu Unterabschnitten, die speziell mit diesem virtuellen Server verbunden sind, befinden sich entlang des oberen Bereichs der Seite. Bevor Sie jedoch irgendeinen der Unterabschnitte konfigurieren, die mit diesem virtuellen Server verbunden sind, stellen Sie diese Seite fertig und klicken Sie auf die Schaltfläche **ACCEPT**.

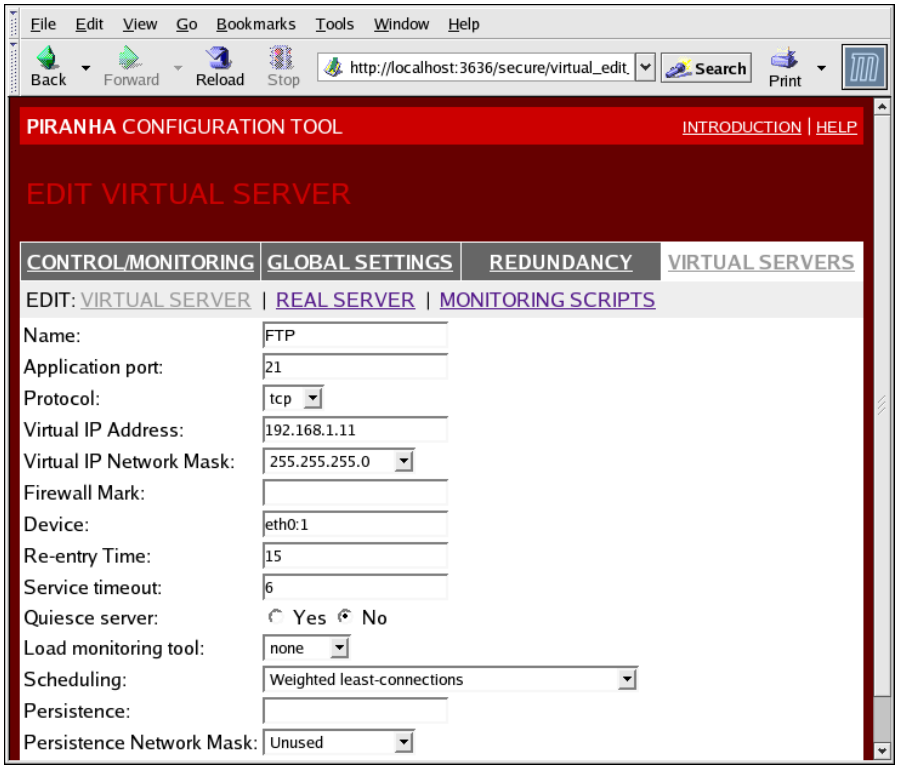

Abbildung 4.6. Der Unterabschnitt VIRTUAL SERVERS

Name

Geben Sie einen aussagekräftigen Namen zur Identifikation des virtuellen Servers an. Dieser Name ist *nicht* der Hostname für die Maschine und sollte daher veranschaulichend und einfach identifizierbar sein. Sie können sogar das Protokoll angeben, das vom virtuellen Server verwendet werden soll, wie zum Beispiel HTTP.

Application port

Geben Sie die Portnummer ein, auf der die Dienst-Applikation horcht. Nachdem dieses Beispiel für HTTP-Dienste gilt, wird Port 80 verwendet.

Protocol

Wählen Sie im Drop-Down-Menü zwischen UDP und TCP. Web-Server kommunizieren typischerweise via TCP-Protokoll, weswegen dies im oben aufgeführten Beispiel ausgewählt ist.

`Virtual IP Address`

Geben Sie die floating IP-Adresse des virtuellen Servers in dieses Textfeld ein.

Virtual IP Network Mask

Setzen Sie die Netzmaske für diesen virtuellen Server mit Hilfe des Drop-Down-Menüs.

`Firewall Mark`

Geben Sie *keinen* ganzzahligen Wert für Firewall-Markierungen in diesem Feld an, bevor Sie nicht Multiport-Protokolle zusammenfassen oder einen virtuellen Multiport-Server für getrennte, jedoch zusammenhängende Protokolle erstellen. In diesem Beispiel besitzt der oben aufgeführte virtuelle Server eine **`Firewall Mark`** von 80, da wir Verbindungen zu HTTP auf Port 80 und zu HTTPS auf Port 443 mit Hilfe des Firewall-Markierungswert 80 zusammenfassen. Zusammen mit Persistenz stellt diese Technik sicher, dass Benutzer, die sowohl auf unsichere, als auch sichere Webseiten zugreifen an denselben realen Server weitergeleitet werden und gleichzeitig der Status beibehalten wird.

Warnung

Die Eingabe einer Firewall-Markierung in diesem Feld ermöglicht es IPVS, dass Pakete, die diese Firewall-Markierung haben, gleich behandelt werden. Sie müssen jedoch weitere Konfiguration außerhalb des **Piranha Configuration Tool** vornehmen, um die Firewall-Markierungen tatsächlich zuzuweisen. Werfen Sie einen Blick auf Abschnitt 3.4, „Multiport-Dienste und das Lastverteilungs-Add-On“ für Anleitungen zur Erstellung von Multi-Port-Diensten und Abschnitt 3.5, „Konfiguration von FTP“ für das Erstellen eines virtuellen Hochverfügbarkeits-FTP-Server.

`Device`

Geben Sie den Namen der Netzwerkschnittstelle, mit der Sie die floating IP-Adresse, die im Feld **`Virtual IP Address`** definiert ist, verknüpft haben möchten.

Sie sollten einen Alias für die öffentliche floating IP-Adresse mit der Ethernet-Schnittstelle, die mit dem privaten Netzwerk verbunden ist, kreieren. In diesem Beispiel befindet sich das private Netzwerk auf der **`eth0`**-Schnittstelle, so dass **`eth0:1`** als der Gerätename eingegeben werden sollte.

`Re-entry Time`

Geben Sie einen ganzzahligen Wert ein, der die Länge der Zeitspanne in Sekunden definiert, bevor der aktive LVS-Router versucht, einen realen Server nach Ausfall wieder in den Pool zu integrieren.

`Service Timeout`

Geben Sie einen ganzzahligen Wert ein, der die Länge der Zeitspanne in Sekunden definiert, bevor der reale Server als inaktiv angesehen und aus dem Pool entfernt wird.

`Quiesce server`

Wenn das Auswahlsymbol **`Quiesce server`** aktiviert ist, wird - jedes Mal, wenn ein neuer realer Server-Knoten online geht - die Tabelle mit den wenigsten Verbindungen auf Null zurückgesetzt, so dass der aktive LVS-Router Anfragen so routet, als ob alle realen Server frisch zum Pool hinzugefügt wurden. Diese Optionen verhindert, dass sich ein neuer Server beim Eintreten in einen Pool durch eine hohe Anzahl an Verbindungen verzettelt.

`Load monitoring tool`

Der LVS-Router kann die Auslastung auf verschiedenen realen Servern entweder mit Hilfe von **`rup`** oder **`ruptime`** überwachen. Falls Sie **`rup`** aus dem Drop-Down-Menü auswählen muss auf jedem realen Server der **`rstatd`**-Dienst laufen. Falls Sie **`ruptime`** auswählen, muss auf jedem realen Server der **`rwhod`**-Dienst laufen.

Warnung

Die Überwachung der Auslastung ist *nicht* dasselbe, wie die Lastverteilung und kann in schwer vorhersagbares Scheduling-Verhalten resultieren, wenn sie mit gewichteten Scheduling-Algorithmen verknüpft wird. Weiterhin müssen die realen Server Linux-Maschinen sein, wenn Sie die Überwachung der Auslastung verwenden.

`Scheduling`

Wählen Sie Ihren bevorzugten Scheduling-Algorithmus aus dem Drop-Down-Menü. Der Standardwert lautet **`Weighted least-connection`**. Werfen Sie einen Blick auf Abschnitt 1.3.1, „Scheduling-Algorithmen“ für weitere Informationen zu Scheduling-Algorithmen.

`Persistence`

Falls ein Administrator persistente Verbindungen während einer Client-Transaktion zum virtuellen Server benötigt, geben Sie die Anzahl der Sekunden von Inaktivität ein, die verstreichen dürfen, bevor eine Verbindung in diesem Textfeld zeitlich abläuft.

Wichtig

Falls Sie in dem oben aufgeführten Feld **`Firewall Mark`** einen Wert eingegeben haben, sollten Sie auch einen Wert für Persistenz eingeben. Stellen Sie außerdem sicher, dass falls Sie Firewall-Markierungen und Persistenz gemeinsam verwenden, die Zahl der Persistenz für jeden virtuellen Server mit der Firewall-Markierung übereinstimmt. Werfen Sie einen Blick auf Abschnitt 1.5, „Persistenz und Firewall-Markierungen“ für weitere Informationen zu Persistenz und Firewall-Markierungen.

`Persistence Network Mask`

Um die Persistenz für ein bestimmtes Subnetz einzuschränken, wählen Sie die entsprechende Netzwerkmaske aus dem Drop-Down-Menü.

Anmerkung

Vor der Einführung von Firewall-Markierungen, war die durch Subnetze eingeschränkte Persistenz eine umständliche Möglichkeit, Verbindungen zusammenzufassen. Gegenwärtig ist es am besten, Persistenz in Verbindung mit Firewall-Markierungen zu verwenden, um dasselbe Ergebnis zu erzielen.

Warnung

Denken Sie nach dem Ändern dieser Seite daran, auf die Schaltfläche **ACCEPT** zu klicken, um sicherzustellen, dass Sie keine Änderungen verlieren, wenn Sie ein neues Panel auswählen.

4.6.2. Der Unterabschnitt REAL SERVER

Ein Klick auf die Verknüpfung des Unterabschnitts **REAL SERVER** im oberen Bereich des Panels zeigt den Unterabschnitt **EDIT REAL SERVER** an. Es zeigt den Status des physikalischen Server-Hosts für einen bestimmten virtuellen Dienst.

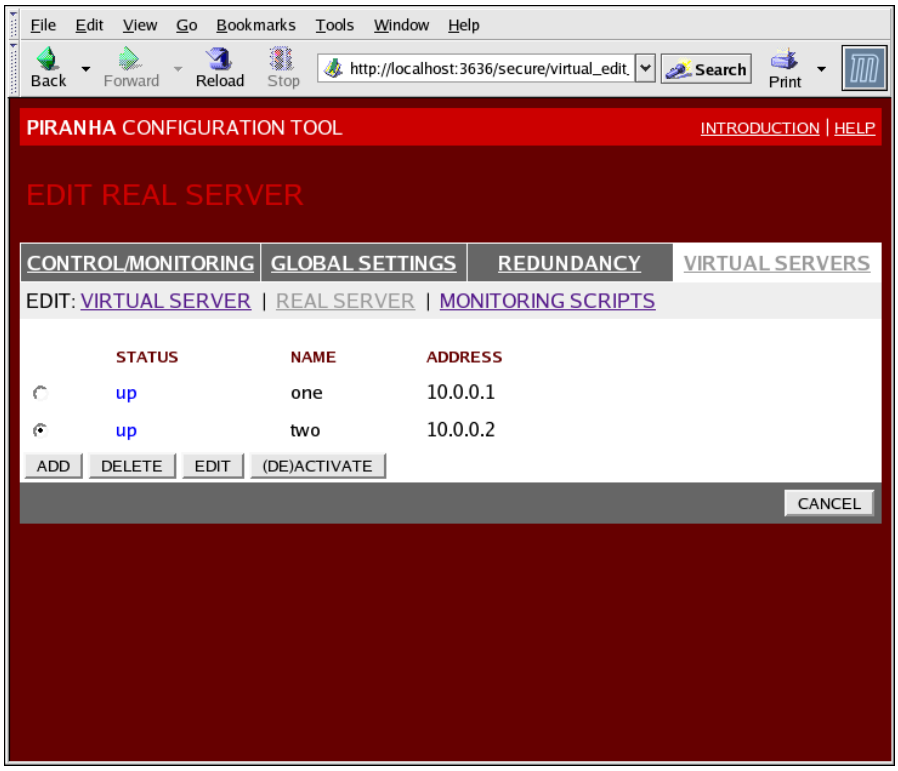

Abbildung 4.7. Der Unterabschnitt REAL SERVER

Klicken Sie auf die Schaltfläche **ADD**, um einen neuen Server hinzuzufügen, wählen die Auswahlschaltfläche daneben aus und klicken auf die Schaltfläche **DELETE**. Klicken Sie auf die Schaltfläche **EDIT**, um das Panel **EDIT REAL SERVER** zu laden, wie unter Abbildung 4.8, „Das

Konfigurationspanel **REAL SERVER**“ dargestellt.

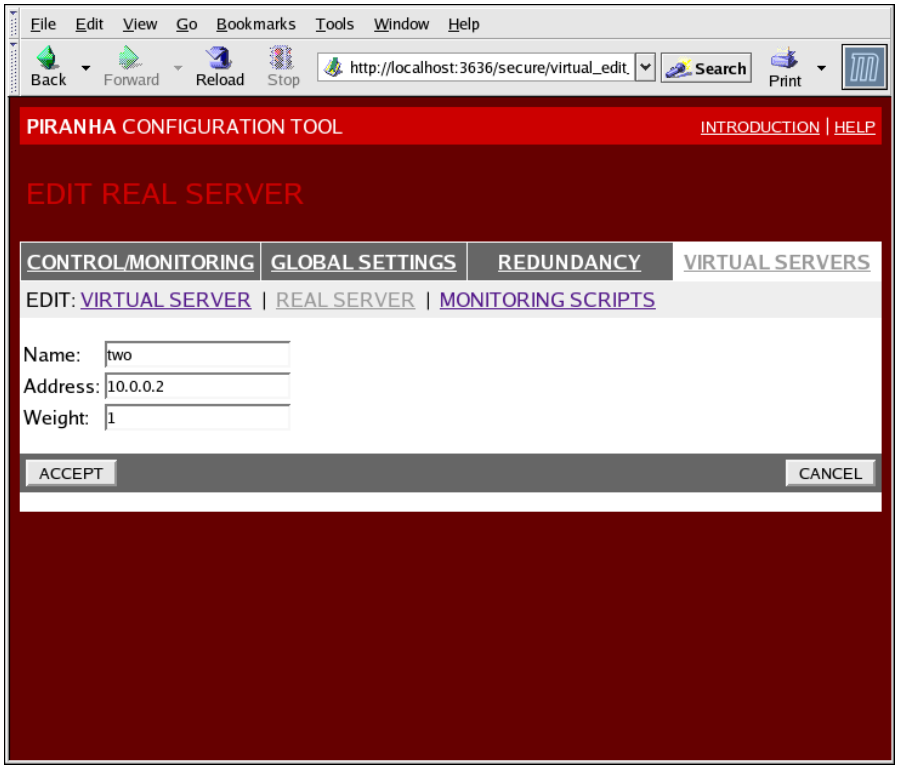

Abbildung 4.8. Das Konfigurationspanel REAL SERVER

Dieses Panel besteht aus drei Eingabefeldern:

`Name`

Ein veranschaulichender Name für den realen Server.

> **Anmerkung**
>
> Dieser Name ist *nicht* der Hostname für die Maschine und sollte daher veranschaulichend und einfach identifizierbar sein.

`Address`

Die IP-Adresse des realen Servers. Nachdem der Port, auf dem gehorcht wird, bereits für den dazugehörigen virtuellen Server angegeben ist, fügen Sie keine Portnummer hinzu.

`Weight`

Ein ganzzahliger Wert, der die Kapazität dieses Hosts relativ zu der Kapazität anderer Hosts im Pool anzeigt. Der Wert kann beliebig sein, sollte jedoch als Kennziffer in Verbindung mit anderen realen Servern behandelt werden. Werfen Sie einen Blick auf Abschnitt 1.3.2, „Server-Gewichtung und Scheduling“ für weitere Informationen zu Server-Gewichtung.

Warnung

Denken Sie nach dem Ändern dieser Seite daran, auf die Schaltfläche **ACCEPT** zu klicken, um sicherzustellen, dass Sie keine Änderungen verlieren, wenn Sie ein neues Panel auswählen.

4.6.3. Der Unterabschnitt EDIT MONITORING SCRIPTS

Klicken Sie auf die Verknüpfung **MONITORING SCRIPTS** am oberen Ende der Seite. Der Unterabschnitt **EDIT MONITORING SCRIPTS** erlaubt dem Administrator die Angabe einer Send/Expect String-Sequenz um zu verifizieren, dass der Dienst für den virtuellen Server auf jedem realen Server funktionsfähig ist. Es ist außerdem der Ort, an dem der Administrator angepasste Skripte angeben kann, um Dienste zu überprüfen, die dynamisch verändernde Daten erfordern.

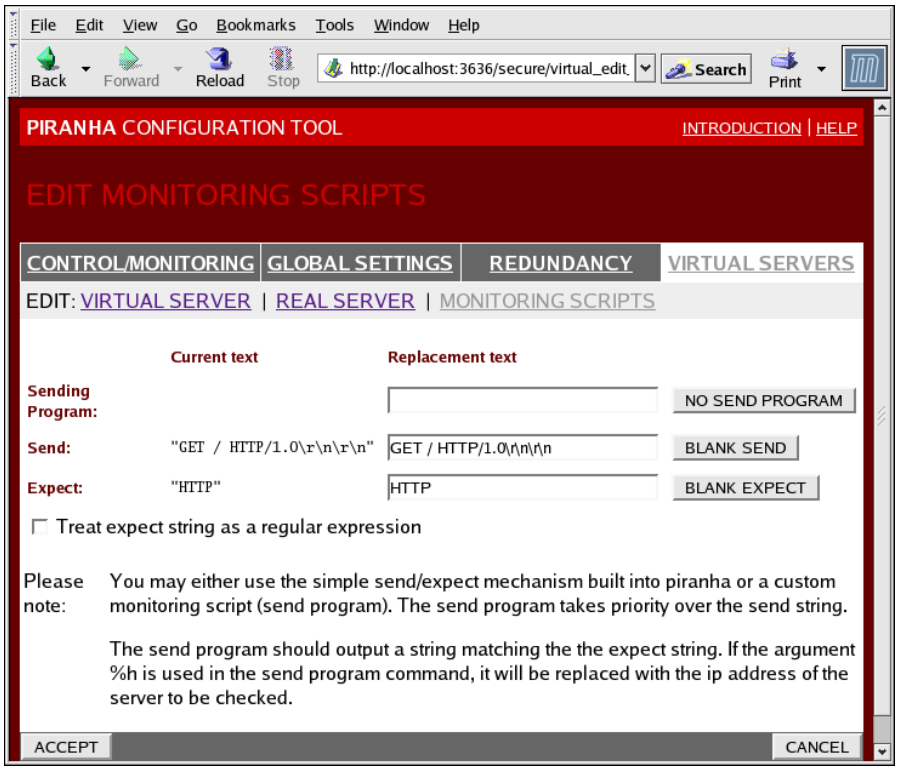
File Edit View Go Bookmarks Tools Window Help
Back Forward Reload Stop
http://localhost:3636/secure/virtual_edit
Search
Print
PIRANHA CONFIGURATION TOOL
INTRODUCTION | HELP
EDIT MONITORING SCRIPTS
CONTROL/MONITORING
GLOBAL SETTINGS
REDUNDANCY
VIRTUAL SERVERS
EDIT: VIRTUAL SERVER | REAL SERVER | MONITORING SCRIPTS
Current text
Replacement text
Sending Program:
NO SEND PROGRAM
Send:
"GET / HTTP/1.0\r\n\r\n"
GET / HTTP/1.0\r\n\r\n
BLANK SEND
Expect:
"HTTP"
HTTP
BLANK EXPECT
Treat expect string as a regular expression
Please note:
You may either use the simple send/expect mechanism built into piranha or a custom monitoring script (send program). The send program takes priority over the send string.
The send program should output a string matching the the expect string. If the argument %h is used in the send program command, it will be replaced with the ip address of the server to be checked.
ACCEPT
CANCEL

Abbildung 4.9. Der Unterabschnitt EDIT MONITORING SCRIPTS

Sending Program

Zur etwas fortgeschrittenen Dienstverifizierung können Sie dieses Feld zur Angabe des Pfades zu einem Skript zur Überprüfung eines Dienstes verwenden. Diese Funktion ist besonders für Dienste hilfreich, die dynamisch verändernde Daten erfordern, wie beispielsweise HTTPS oder SSL.

Um diese Funktion zu verwenden, müssen Sie ein Skript schreiben, das eine Antwort in Textform zurückgibt, dieses ausführbar machen und den Pfad im Feld **Sending Program** eingeben.

Anmerkung

Um sicherzustellen, dass jeder Server im Pool der realen Server überprüft wird, verwenden Sie den speziellen Token **%h**, im Anschluss an den Pfad im Feld **Sending Program**. Dieser Token wird durch die IP-Adresse eines jeden realen Servers ersetzt, wenn das Sktipt vom **nanny**-Daemon aufgerufen wird.

Nachfolgend ist ein Beispiel-Skript aufgeführt, das bei der Zusammenstellung eines externen Skript zur Überprüfung von Diensten als Vorlage verwendet werden kann:

```
#!/bin/sh

TEST=`dig -t soa example.com @$1 | grep -c dns.example.com

if [ $TEST != "1" ]; then
 echo "OK
else
 echo "FAIL"
fi
```

Anmerkung

Falls im Feld **Sending Program** ein externes Programm eingegeben wird, dann wird das Feld **Send** ignoriert.

Send

Geben Sie einen String für den **nanny**-Daemon ein, der an jeden realen Server in diesem Feld gesendet werden soll. Standardmäßig wird das Send-Feld für HTTP vervollständigt. Sie können diesen Wert abhängig von Ihren Anforderungen ändern. Wenn Sie dieses Feld leer lassen, versucht der **nanny**-Daemon, den Port zu öffnen und geht davon aus, dass der Dienst läuft, wenn er erfolgreich ist.

Nur eine Send-Sequenz ist in diesem Feld gestattet und es kann nur druckbare, ASCII-Zeichen, sowie die folgenden Code-Umschaltzeichen enthalten:

- \n für Zeilenumbruch.
- \r für Wagenrücklauf.
- \t für Tabulator.
- \ für den Escape des nächsten Zeichens, das darauf folgt.

Expect

Geben Sie eine Antwort in Textform ein, die der Server zurückgeben soll, falls er ordnungsgemäß funktioniert. Falls Sie Ihr eigenes Programm zum Versenden geschrieben haben, geben Sie die Antwort ein, die sie ihm im Falle eines Erfolgs zugewiesen haben.

Anmerkung

Um zu ermitteln, was für einen vorgegebenen Dienst versendet werden soll, können Sie eine **telnet**-Verbindung mit dem Port auf dem realen Server herstellen und so sehen, was dieser zurückgibt. So gibt FTP beispielsweise 220 bei der Verbindung zurück, so dass Sie **quit** im Feld **Send** und **220** im Feld **Expect** eingeben könnten.

Warnung

Denken Sie nach dem Ändern dieser Seite daran, auf die Schaltfläche **ACCEPT** zu klicken, um sicherzustellen, dass Sie keine Änderungen verlieren, wenn Sie ein neues Panel auswählen.

Sobald Sie die virtuellen Server mit dem **Piranha Configuration Tool** konfiguriert haben, müssen Sie spezifische Konfigurationsdateien auf den Backup-LVS-Router kopieren. Werfen Sie einen Blick auf Abschnitt 4.7, „Synchronisation von Konfigurationsdateien“ für weitere Details.

4.7. Synchronisation von Konfigurationsdateien

Nach der Konfiguration des primären LVS-Routers gibt es mehrere Konfigurationsdateien, die auf den Backup-LVS-Router kopiert werden müssen, bevor Sie das Lastverteilungs-Add-On starten können.

Diese Dateien umfassen:

- **/etc/sysconfig/ha/lvs.cf** — Die Konfigurationsdatei für die LVS-Router.
- **/etc/sysctl** — Die Konfigurationsdatei, die neben anderen Dingen die Paketweiterleitung im Kernel aktiviert.
- **/etc/sysconfig/iptables** — Falls Sie Firewall-Markierungen verwenden, sollten Sie eine dieser Dateien synchronisieren, abhängig davon, welchen Paketfilter Sie verwenden.

Wichtig

Die Dateien **/etc/sysctl.conf** und **/etc/sysconfig/iptables** ändern sich nicht, wenn Sie das Lastverteilungs-Add-On mit dem **Piranha Configuration Tool** konfigurieren.

4.7.1. Synchronisation von `lvs.cf`

Jedes Mal, wenn die LVS-Konfigurationsdatei **/etc/sysconfig/ha/lvs.cf** erstellt oder aktualisiert wird, müssen Sie sie auf den Backup-LVS-Router-Knoten kopieren.

Warnung

Sowohl der aktive, als auch der Backup-LVS-Router-Knoten muss identische **lvs.cf**-Dateien besitzen. LVS-Konfigurationsdateien unter den LVS-Router-Knoten können eine Ausfallsicherung verhindern.

Am besten kann dies mit **scp** durchgeführt werden.

Wichtig

Zur Verwendung von **scp** muss **sshd** auf dem Backup-Router laufen. Werfen Sie einen Blick auf Abschnitt 2.1, „Konfiguration von Diensten auf den LVS-Routern“ für Details zur ordnungsgemäßen Konfiguration der notwendigen Dienste auf den LVS-Routern.

Führen Sie den folgenden Befehl als Root-Benutzer auf dem primären LVS-Router aus, um die **lvs.cf**-Dateien zwischen den Router-Knoten zu synchronisieren:

```
scp /etc/sysconfig/ha/lvs.cf n.n.n.n:/etc/sysconfig/ha/lvs.cf
```

Ersetzen Sie in diesem Befehl ***n.n.n.n*** mit der realen IP-Adresse des Backup-LVS-Routers.

4.7.2. Synchronisation von sysctl

Die **sysctl**-Datei wird in den meisten Situationen nur einmal modifiziert. Diese Datei wird zum Zeitpunkt des Bootens gelesen und weist den Kernel an, Paketweiterleitung zu aktivieren.

Wichtig

Falls Sie sich nicht sicher sind, ob die Paketweiterleitung im Kernel aktiviert ist, oder nicht, werfen Sie einen Blick auf Abschnitt 2.5, „Aktivierung der Paketweiterleitung“ für Anweisungen, wie Sie diese Kernfunktion überprüfen und ggf. aktivieren können.

4.7.3. Synchronisation von Regeln zur Netzwerk-Paket-Filterung

Falls Sie **iptables** verwenden, müssen Sie die entsprechende Konfigurationsdatei auf dem Backup-LVS-Router synchronisieren.

Falls Sie irgendeine Netzwerk-Paketfilter-Regel verändern, geben Sie den folgenden Befehl als Root auf dem primären LVS-Router ein:

```
scp /etc/sysconfig/iptables n.n.n.n:/etc/sysconfig/
```

Ersetzen Sie in diesem Befehl ***n.n.n.n*** mit der realen IP-Adresse des Backup-LVS-Routers.

Öffnen Sie als nächstes entweder eine **ssh**-Sitzung zum Backup-Router oder loggen Sie sich als Root auf der Maschine ein und geben folgenden Befehl ein:

```
/sbin/service iptables restart
```

Sobald Sie diese Dateien auf den Backup-Router kopiert haben und die entsprechenden Dienste (werfen Sie einen Blick auf Abschnitt 2.1, „Konfiguration von Diensten auf den LVS-Routern“ für mehr zu diesem Thema) gestartet haben, sind Sie bereit, das Lastverteilungs-Add-On zu starten.

4.8. Das Lastverteilungs-Add-On starten

Um das Lastverteilungs-Add-On zu starten, eignen sich zwei simultan geöffnete Root-Terminals oder zwei von Root simultan geöffnete **ssh**-Sitzungen zum primären LVS-Router am besten.

Überwachen Sie in einem Terminal die Protokollmeldungen des Kernels mit dem Befehl:

```
tail -f /var/log/messages
```

Starten Sie anschließend das Lastverteilungs-Add-On mit dem folgenden Befehl in dem anderen Terminal:

```
/sbin/service pulse start
```

Verfolgen Sie den Verlauf des Startprozesses des **pulse**-Dienstes in dem Terminal mit den Protokollmeldungen des Kernels. Wenn Sie folgende Ausgabe sehen, wurde der pulse-Daemon ordnungsgemäß gestartet:

```
gratuitous lvs arps finished
```

Drücken Sie **Strg**+**c**, um das Beobachten von **/var/log/messages** zu beenden.

Ab diesem Punkt ist der primäre LVS-Router auch der aktive LVS-Router. Auch wenn an diesem Punkt bereits Anfragen an LVS gestellt werden können, sollten Sie den Backup-LVS-Router starten, bevor Sie das Lastverteilungs-Add-On in Betrieb nehmen. Führen Sie hierfür einfach den oben beschriebenen Prozess für den Backup-LVS-Router-Knoten erneut durch.

Nach Abschluss des letzten Schritts, ist das Lastverteilungs-Add-On aktiv und läuft.

Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On

Sie können das Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On verwenden, um eine hochverfügbare E-Commerce-Site einzusetzen, die Lastverteilung, Datenintegrität und Verfügbarkeit von Applikationen bietet.

Die Konfiguration in Abbildung A.1, „Lastverteilungs-Add-On mit einem Hochverfügbarkeits-Add-On“ repräsentiert eine E-Commerce-Site, die für Online-Bestellung von Waren mit Hilfe einer URL verwendet wird. Anfragen von Clients an die URL passieren die Firewall zum aktiven LVS Lastverteilungs-Router, welcher diese dann an einen der Web-Server weiterleitet. Die Hochverfügbarkeits-Add-On-Knoten stellen den Web-Servern dynamische Daten zur Verfügung, welche diese dann an die anfragenden Clients weiterleiten.

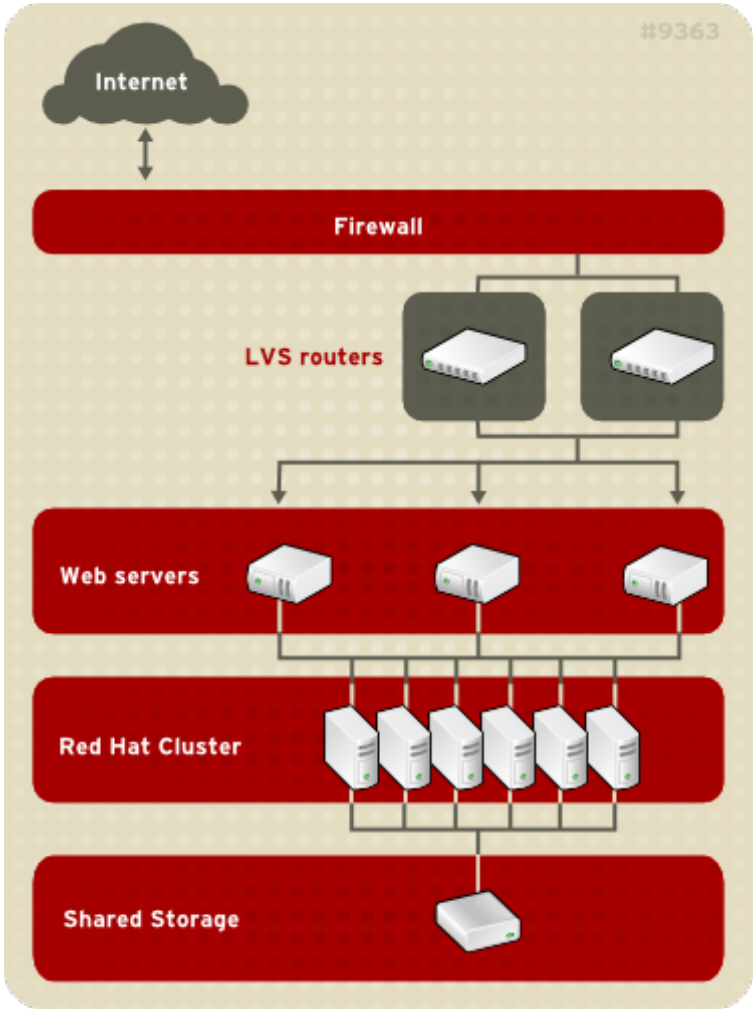

Abbildung A.1. Lastverteilungs-Add-On mit einem Hochverfügbarkeits-Add-On

Das Bereitstellen von dynamischen Web-Inhalt erfordert eine Three-Tier-Konfiguration (wie unter Abbildung A.1, „Lastverteilungs-Add-On mit einem Hochverfügbarkeits-Add-On“ gezeigt). Diese Kombination des Lastverteilungs-Add-Ons und der Hochverfügbarkeits-Add-Ons ermöglicht die Konfiguration einer hoch-integrativen, "no-single-point-of-failure" E-Commerce-Site. Das Hochverfügbarkeits-Add-On kann eine Hochverfügbarkeitsinstanz einer Datenbank oder einer Reihe von Datenbanken, auf die von den Web-Servern via Netzwerk zugegriffen werden kann, ausführen.

Eine Three-Tier-Konfiguration wird benötigt, um dynamischen Inhalt zur Verfügung zu stellen. Auch

wenn eine Two-Tier Lastverteilungs-Add-On-Konfiguration für die Bereitstellung von statischem Web-Inhalt (bestehend aus einer geringen Menge von sich selten verändernden Daten) durch Web-Server geeignet ist, ist sie nicht dafür geeignet, wenn die Web-Server dynamischen Inhalt bereitstellen. Dynamischer Inhalt könnte Warenbestand, Bestellungen oder Kundendatenbanken beinhalten, welche auf allen Web-Servern einheitlich sein müssen, um zu gewährleisten, dass Kunden Zugriff auf aktuelle und präzise Informationen hat.

Jede Tier liefert die folgenden Funktionen:

- Erste Tier — LVS-Router, die Lastverteilung durchführen, um Web-Anfragen zu verteilen.
- Zweite Tier — Eine Reihe von Web-Servern, die die Anfragen bearbeiten.
- Dritte Tier — Ein Hochverfügbarkeits-Add-On, das Daten für die Web-Server liefert.

In einer Hochverfügbarkeits-Add-On-Konfiguration wie der in Abbildung A.1, „Lastverteilungs-Add-On mit einem Hochverfügbarkeits-Add-On“, stellen Clients Anfragen innerhalb des World Wide Web. Aus Sicherheitsgründen gehen diese Anfragen via Firewall bei einer Web-Site ein, welche ein für diesen Zweck gedachtes Linux-System, oder ein dediziertes Firewall-Gerät sein kann. Aus Gründen der Redundanz können Sie Firewall-Geräte in einer Konfiguration zur Ausfallsicherung konfigurieren. Hinter der Firewall befinden sich LVS-Router für die Lastverteilung, welche in einem "active-standby" Modus konfiguriert werden können. Der aktive Lastverteilungs-Router leitet die Anfragen an die Gruppe von Web-Servern weiter.

Jeder Web-Server kann eine HTTP-Anfrage von einem Client unabhängig verarbeiten und die Antwort zurück an den Client schicken. Das Hochverfügbarkeits-Add-On ermöglicht Ihnen, die Kapazität einer Web-Site zu erweitern, indem Web-Server hinter den LVS-Routern hinzugefügt werden. Die LVS-Router führen dann eine Lastverteilung unter einer größeren Gruppe von Web-Servern durch. Zusätzlich kann ein Web-Server bei einem Ausfall entfernt werden. Das Hochverfügbarkeits-Add-On fährt dann damit fort, die Lastverteilung unter einer kleineren Gruppe von Web-Servern durchzuführen.

Revisionsverlauf

Version 6-3.400	**2013-10-31**	**Rüdiger Landmann**
Rebuild with publican 4.0.0		
Version 6-3	**2012-07-18**	**Anthony Towns**
Rebuild for Publican 3.0		
Version 1.0-0	**Wed Nov 10 2010**	**Paul Kennedy**
Erstmalige Veröffentlichung des Buches für Red Hat Enterprise Linux 6		

Stichwortverzeichnis

Symbole

/etc/sysconfig/ha/lvs.cf file, /etc/sysconfig/ha/lvs.cf

A

arptables_jf, Direktes Routing und arptables_jf

C

chkconfig, Konfiguration von Diensten auf den LVS-Routern

Cluster
- Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On, Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On

D

Direktes Routing
- und arptables_jf, Direktes Routing und arptables_jf

E

Einführung, Einführung
- Weitere Red Hat Enterprise Linux Dokumente, Einführung

F

Feedback, Feedback

FTP, Konfiguration von FTP
- (Siehe auch Lastverteilungs-Add-On)

G

Gewichtete Least-Connections (Siehe Job-Scheduling, Lastverteilungs-Add-On)

Gewichtetes Round-Robin (Siehe Job-Scheduling, Lastverteilungs-Add-On)

H

Hochverfügbarkeits-Add-On
- und das Lastverteilungs-Add-On , Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On
- Verwendung des Lastverteilungs-Add-Ons mit , Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On

I

iptables , Konfiguration von Diensten auf den LVS-Routern

ipvsadm program, ipvsadm

J

Job-Scheduling, Lastverteilungs-Add-On, Lastverteilungs-Add-On Scheduling-Überblick

K

Komponenten
- des Lastverteilungs-Add-Ons, Komponenten des Lastverteilungs-Add-Ons

L

Lastverteilungs-Add-On
- /etc/sysconfig/ha/lvs.cf file, /etc/sysconfig/ha/lvs.cf
- das Lastverteilungs-Add-On starten, Das Lastverteilungs-Add-On starten
- Datenaustausch, Datenreplikation und Datenaustausch zwischen realen Servern
- Datenreplikation, Reale Server, Datenreplikation und Datenaustausch zwischen realen Servern
- Direktes Routing
  - Anforderungen, Hardware, Direktes Routing, Das Lastverteilungs-Add-On via direktem Routing
  - Anforderungen, Netzwerk, Direktes Routing, Das Lastverteilungs-Add-On via direktem Routing
  - Anforderungen, Software, Direktes Routing, Das Lastverteilungs-Add-On via direktem Routing
  - und arptables_jf, Direktes Routing und arptables_jf
- Erstmalige Konfiguration, Erstmalige Konfiguration des Lastverteilungs-Add-Ons
- ipvsadm Programm, ipvsadm
- Job-Scheduling, Lastverteilungs-Add-On Scheduling-Überblick
- Komponenten von, Komponenten des Lastverteilungs-Add-Ons
- LVS-Router
  - Benötigte Dienste, Konfiguration von Diensten auf den LVS-Routern
  - Dienste konfigurieren, Erstmalige Konfiguration des Lastverteilungs-Add-Ons
  - Primäre Knoten, Erstmalige Konfiguration des Lastverteilungs-Add-Ons
- Multiport-Dienste, Multiport-Dienste und das Lastverteilungs-Add-On

- FTP, Konfiguration von FTP

- nanny Daemon, nanny
- NAT-Routing
  - Anforderungen, Hardware, Das NAT Lastverteilungs-Add-On-Netzwerk
  - Anforderungen, Netzwerk, Das NAT Lastverteilungs-Add-On-Netzwerk
  - Anforderungen, Software, Das NAT Lastverteilungs-Add-On-Netzwerk

- Paketweiterleitung, Aktivierung der Paketweiterleitung
- Piranha Configuration Tool , Piranha Configuration Tool
- pulse Daemon, pulse
- Routing-Methoden
  - NAT, Routing-Methoden

- Routing-Voraussetzungen, Konfiguration von Netzwerkschnittstellen für das Lastverteilungs-Add-On mit NAT
- Scheduling, Job, Lastverteilungs-Add-On Scheduling-Überblick
- send_arp Programm, send_arp
- Synchronisation von Konfigurationsdateien, Synchronisation von Konfigurationsdateien
- Three-Tier
  - Red Hat Cluster Manager, Eine Three-Tier-Lastverteilungs-Add-On-Konfiguration

- Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On, Verwendung des Lastverteilungs-Add-On mit dem Hochverfügbarkeits-Add-On

Least-Connections (Siehe Job-Scheduling, Lastverteilungs-Add-On)

LVS

- Daemon, lvs
- lvs Daemon, lvs
- NAT-Routing
  - aktivieren, Aktivierung von NAT-Routing auf den LVS-Routern

- Reale Server, Überblick über das Lastverteilungs-Add-On
- Überblick über, Überblick über das Lastverteilungs-Add-On

lvs daemon, lvs

M

Multiport-Dienste, Multiport-Dienste und das Lastverteilungs-Add-On
- (Siehe auch Lastverteilungs-Add-On)

N

nanny daemon, nanny

NAT

- aktivieren, Aktivierung von NAT-Routing auf den LVS-Routern
- Routing-Methoden, Lastverteilungs-Add-On, Routing-Methoden

Network Address Translation (Siehe NAT)

P

Paketweiterleitung, Aktivierung der Paketweiterleitung
- (Siehe auch Lastverteilungs-Add-On)

Piranha Configuration Tool , Piranha Configuration Tool
- benötigte Software, Benötigte Software
- CONTROL/MONITORING , CONTROL/MONITORING
- EDIT MONITORING SCRIPTS Unterabschnitt, Der Unterabschnitt EDIT MONITORING SCRIPTS
- Ein Passwort setzen, Einrichten eines Passworts für das Piranha Configuration Tool
- GLOBAL SETTINGS , GLOBAL SETTINGS
- Login-Panel, Einloggen in das Piranha Configuration Tool
- REAL SERVER Unterabschnitt, Der Unterabschnitt REAL SERVER
- REDUNDANCY , REDUNDANCY
- Überblick über, Konfiguration des Lastverteilungs-Add-Ons mit dem Piranha Configuration Tool
- VIRTUAL SERVER Unterabschnitt
    - Firewall Mark , Der Unterabschnitt VIRTUAL SERVER
    - Persistence , Der Unterabschnitt VIRTUAL SERVER
    - Scheduling , Der Unterabschnitt VIRTUAL SERVER
    - Virtual IP Address , Der Unterabschnitt VIRTUAL SERVER

- VIRTUELLE SERVER , VIRTUAL SERVERS
- VIRTUELLE SERVER Unterabschnitt, Der Unterabschnitt VIRTUAL SERVER
- Zugriff einschränken, Zugangsbeschränkung zum Piranha Configuration Tool

piranha-gui-Dienst, Konfiguration von Diensten auf den LVS-Routern

piranha-passwd , Einrichten eines Passworts für das Piranha Configuration Tool

pulse daemon, pulse

pulse-Dienst, Konfiguration von Diensten auf den LVS-Routern

R

Reale Server
- Konfiguration von Diensten, Konfiguration von Diensten auf den realen Servern

Round Robin (Siehe Job-Scheduling, Lastverteilungs-Add-On)

Routing
- Voraussetzungen für das Lastverteilungs-Add-On, Konfiguration von Netzwerkschnittstellen für das Lastverteilungs-Add-On mit NAT

S

Scheduling, Job (Lastverteilungs-Add-On), Lastverteilungs-Add-On Scheduling-Überblick

security

- Piranha Configuration Tool , Zugangsbeschränkung zum Piranha Configuration Tool

send_arp program, send_arp

sshd-Dienst, Konfiguration von Diensten auf den LVS-Routern

Synchronisation von Konfigurationsdateien, Synchronisation von Konfigurationsdateien